

POLITECNICO DI MILANO

Scuola di Ingegneria
Industriale e dell'Informazione

Dipartimento di Ingegneria Energetica



IMPACT OF REAL GAS EFFECTS IN MULTISTAGE TURBINE OPTIMIZATION

Relatore: Prof. Marco Astolfi

Tesi di laurea di:

Fabio Ciardullo Matr. 833417

Anno accademico 2015/2016

IMPACT OF REAL GAS EFFECTS IN MULTISTAGE TURBINE OPTIMIZATION

Ringraziamenti

Vorrei ringraziare il mio relatore Prof. Marco Astolfi per la disponibilità e per tutto il materiale e le spiegazioni che mi ha fornito per aiutarmi a stendere questa tesi.

Un particolare ringraziamento al Dr. Luca Moretti per avermi aiutato a comprendere il funzionamento di axtur e per la sua disponibilità nell'aiutarmi.

Un ringraziamento alla mia famiglia che mi ha sempre supportato e mi ha spronato a continuare e arrivare fino in fondo a questo percorso.

In fine ringrazio tutti i miei amici e parenti per avermi aiutato a sopportare questi anni di studio.

Abstract

To analyze the real gas effects in the optimization of a multistage axial turbine a program for the calculation of the thermodynamic properties of fluids (RFEPROP) is integrated in a program for the optimization of multistage gas turbines (Axtur). After a series of tests and validations obtained by simulating fluid that well approximate the behavior of ideal gasses, a cycle is analyzed with a hydrocarbon for ORC applications (Organic Rankine Cycle). Simulating an expansion that starts very close to the saturation line, where strong real gas effects are present, an analysis of the differences that the optimization process is carried out. The new Axtur proves a reliable and easy to use tool to simulate the behavior of real gases in the optimization process of a turbomachine.

Keywords: Real gas, Optimization, Turbine, Axtur.

Sintesi

Per analizzare gli effetti di gas reale nell'ottimizzazione di una turbina assiale multistadio è stato integrato un programma per il calcolo delle proprietà termodinamiche dei fluidi (REFPROP) in un programma per l'ottimizzazione di turbine a gas multistadio (AxTur). Dopo una serie di analisi e validazioni ottenute simulando fluidi che approssimino bene il comportamento dei gas ideali, viene analizzato un ciclo con un idrocarburo per applicazioni ORC (Organic rankine cycle). Simulando un'espansione che inizia in un punto prossimo alla linea di saturazione con forti effetti di gas reale, vengono analizzate le differenze che il processo di ottimizzazione deve seguire per cercare di adattare la turbomacchina ad un fluido che si discosta molto da un gas ideale. Il nuovo AxTur si rivela uno strumento affidabile e di facile utilizzo per simulare il comportamento dei gas reali nel processo di ottimizzazione di una turbomacchina.

Parole chiave: Gas reale, Ottimizzazione, Turbina, Axtur.

Index

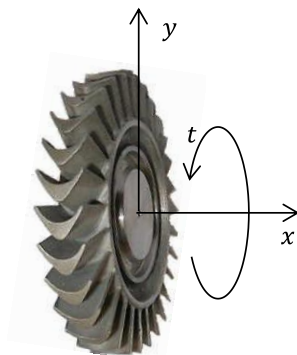
1. Introduction
 - 1.1. Design of axial turbines
 - 1.2. 0-1-2-3 D models
 - 1.3. The process of optimization
2. Axtur
 - 2.1. Optimization process
 - 2.2. Algorithm and flowchart
 - 2.3. Validation of the conversion
 - 2.4. Analysis of optimization functions and algorithms
3. New features
 - 3.1. Ideal vs real gas
 - 3.1.1. Neon
 - 3.1.2. Nitrogen
 - 3.1.3. Octane
 - 3.2. Effect increasing number of stages
4. Conclusions
 - 4.1. Further development

1. Introduction

Axial-flow turbines are the most widely spread turbomachinery adopted for electric power generation. Due to their key importance in a power plant's operating costs, maximization of the efficiency has always represented the main goal of a design process. Here are discussed the main characteristics of an optimization problem and the assumption needed to describe in a simplified, yet as accurate as possible, way the complexity of the fluido-thermodynamic process that take place in its cascades.

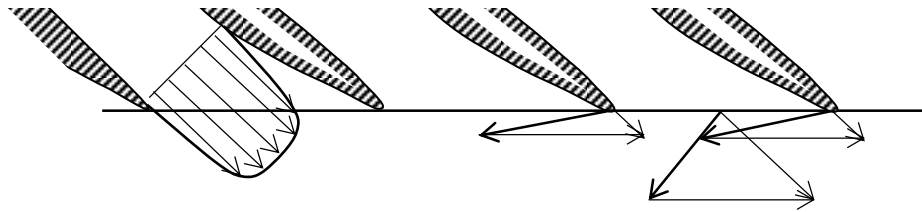
1.1 Design of axial turbines

The fluid dynamics process occurring in a turbomachine could be completely define by the fluid motion or Navier-Stokes equations. Although these equations have been known for centuries, their numerical solution could be extremely difficult to achieve. A three dimensional approach is needed to completely determine the flows and the thermodynamic quantities within a turbomachine. Axial, Radial and Tangential are the three most adopted coordinates to describe the process. The first two determine the meridian plane while the first and the third determine the "blade-to-blade" surface.



Coordinate system for an axial-flow turbine blade row

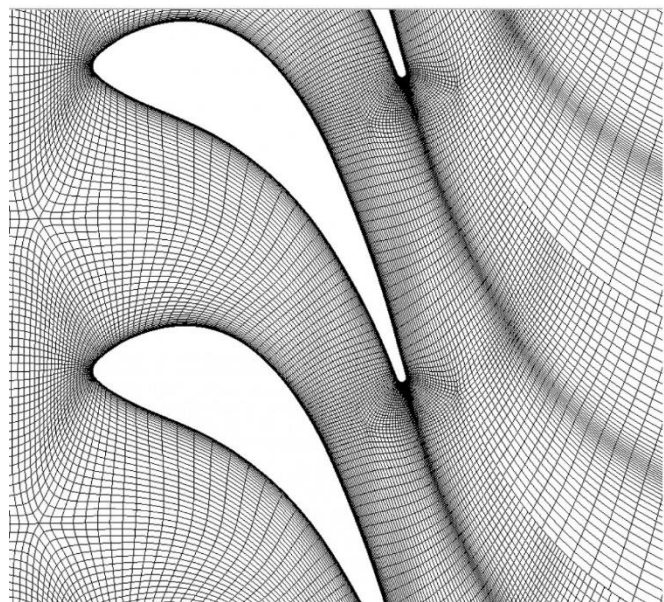
Any simplification to the process is prevented by irreversible effects (friction, turbulence, etc.) that can vastly affect the solution if are not taken in to account carefully. The matter is further complicated by non-stationarity of the process that causes pulsation and instability of the flow.



Velocity profile trend within the inter-blade channel

As the fluid comes out from the rotating blades, the velocity assumes a pulsating trend due to the thickness of the blades that need to be designed to withstand fatigue during the operating life of the machine.

The geometry of a turbomachine is very complex: to carefully describe the process in the vicinity of the walls very small mesh need to be adopted, leading to an excessive number of computing cells. “Turbulence models” need to be applied to simplify the process, but due to the intrinsic randomness of turbulent fluid motions, they require a fine calibration on a case-by-case basis. To further complicate the matter the scale of processes occurring in the machine varies of many orders of magnitude, from interblade channel to radial seal labyrinths, requiring a variable dimension mesh to carefully describe them. Also in this case simplified models are adopted to limit



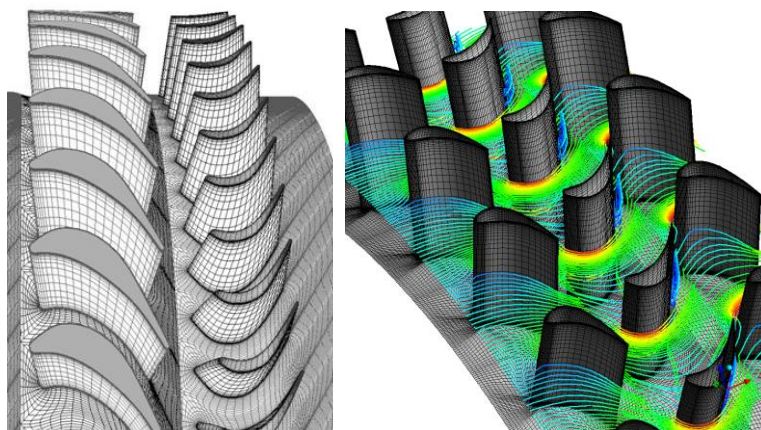
the computational workload involved.

1.2 1-2-3 Dimensions Models

Different approaches can be developed to describe the flow inside a turbomachine.

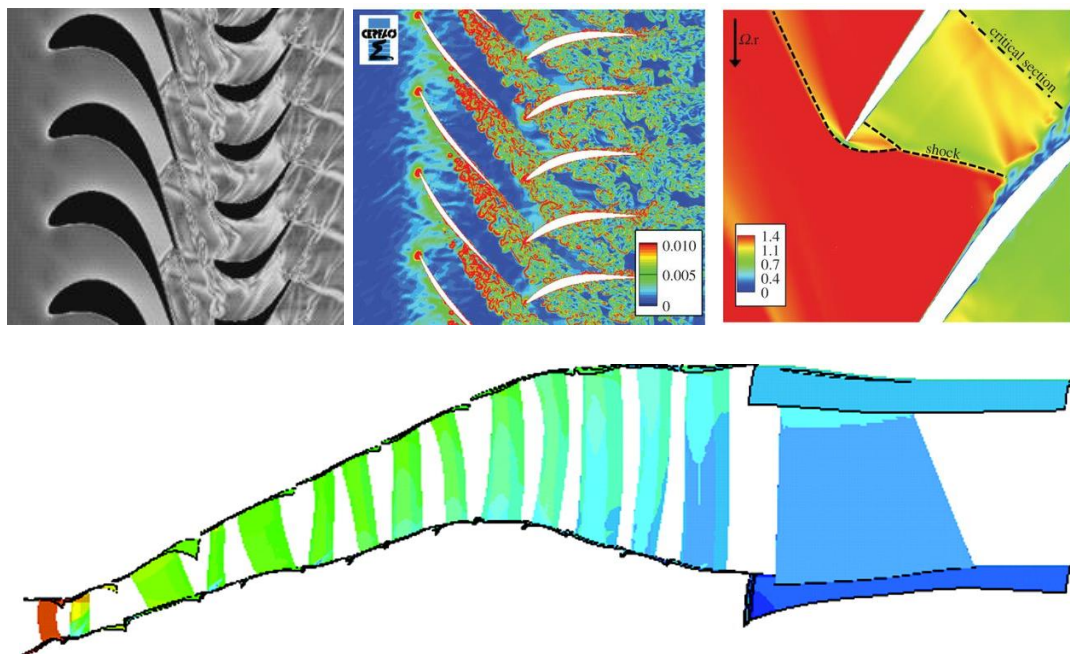
Three-dimensional: Through a complete solution of the flow field that would allow the identification of three-dimensional turbulence phenomena, such as vortices, streaks, three-dimensional shock waves, vortex shedding, etc. This approach is extremely costly in terms of computational power, and without the appropriate simplifications mainly related to the turbulence models, cannot be tackled numerically.

In conclusion, it is extremely difficult to calculate the equations on the entire three-dimensional field due to complex geometries, even if nowadays increasingly refined turbulence models and the availability of an ever greater computing power allow to reach a level of detail that was unthinkable just a few years ago. Either way, it is still not possible to obtain numerical solutions independent of simplifications and calibrations obtained from experimental tests and previously known solutions.



Examples of three-dimensional mesh for the study of an axial turbine stage

Two-dimensional: Always very complex from a computational point of view. The choice of the section on which to carry out the analysis can fall within a "meridian" or more commonly on the "blade-to-blade" plane. Currently, most of the CFD research is performed using the "blade-to-blade" plane since it is the section that allows the identification of the phenomena of vortex shedding, shock waves, etc. and that better characterizes the efficiency of a stage. There should, however, be loss correlations to account for the three-dimensionality of the blade.



Examples of CFD studies in a two-dimensional "blade-to-blade" plane and in the meridional plane.

One-dimensional in which the characteristics of the fluid are assessed on an upstream and downstream media section of each blade row. In this case it is necessary to rely upon experimental loss correlations to characterize the performance of a given row, starting from its geometric dimensions and the typology of flow (subsonic or supersonic)

Each of these approaches, with their respective potential and limits, allows to evaluate the performance of a blade row of a turbine stage and therefore, more in general, the possibility of evaluating the performance of multistage machines.

1.3 The process of optimization

In the particular case in which we do not want to know the performance of a known geometry turbine but want to optimize a turbine processing a certain flow rate of fluid with given input conditions and exhaust pressure, we will need to evaluate the performance of many turbines in order to identify the best one. A modern approach, which allows the use of considerable computing power for the design of a turbomachine, leads to formulate the problem as a multi-variable optimization problem in the presence of constraints. It is indispensable, on one side, to reduce the number of variables in order to limit the number of iterations necessary to achieve the optimum, and on the other to reduce the computational time of each single iteration. In this respect, it is clear that a three-dimensional or two-dimensional approach is too costly, since each iteration requires the resolution of the flow field and leads to an exponential increase in computing time. The commonly followed approach is that of the one-dimensional description of the machine, in order to reduce the optimization variables and expedite the efficiency evaluation with the use of loss correlations validated on experimental cases. In any case, as will be seen in the following, even with a one-dimensional approach the number of geometric variables for which a turbomachine can be optimized already makes the problem numerically complex and requires powerful and robust algorithms to address it.

The choice to adopt a one-dimensional approach, however, should not be considered as a fall-back solution due to limits in computing power or in the theoretical description of certain phenomena, because the optimal design obtained by this method, allows a sufficiently accurate definition of the geometry of the machine, the rotation velocity and optimal velocity triangles.

Once the theoretical maximum performance of a stage is defined, optimizing each individual geometric variable and taking exactly into account every dissipative phenomenon, a one-dimensional approach allows to reach more than 95% of this limit defining correctly the design choices. Using more accurate CFD methods, it will be possible to obtain a further increase in efficiency that undoubtedly will make a difference in terms of power production but, however, will never make up for incorrect choices possibly made during the previous phase.

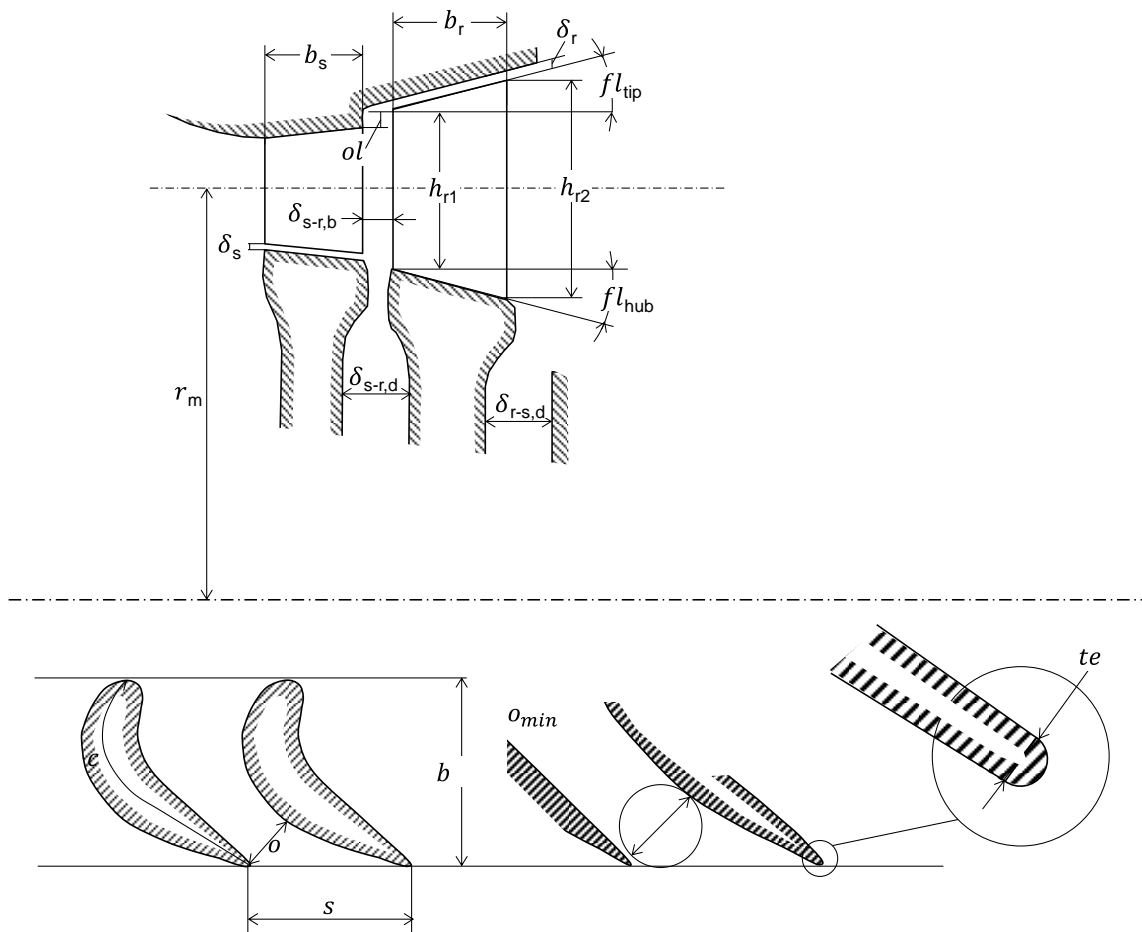
In an optimization problem of this type it is always necessary to define:

- *The objective function*: it may be the efficiency to be maximized (in its various formulations), or costs, or the dimensions, or (more rationally) a function which assigns different *weights* to these parameters.
- *Independent variables* to be used in the optimal value research: depending on the approach used to describe the turbine, it will be necessary to identify the variables for a complete description of the stage. The simplest approach is the one-dimensional, which will be followed here, in which around ten variables are identified for each single stage. For multistage machines, this leads to a quite wide dimension of the field of solutions and to considerable computational costs, even if the time of each single iteration is minimized through the use of appropriate correlations.
- *limits* can be conceptually classified into two categories:
 - Limits that take into account the technological and/or economic constraints in the construction phase: for example, an upper limit to the number of blades in a row, or a minimum limit to gaps and thicknesses. They can be absolute or related to another quantity.
 - Limits that maintain the field of investigation within a range of solutions for which adequate correlations for the prevision of the losses are available and/or that allow to rule out solutions that are unavoidably characterized by low efficiencies: for example, a maximum limit to the angular deviation in a blade row, or to Mach numbers.

- *limits* can also be divided into:

- Linear limits, which define the interval between a minimum and a maximum of optimization variables within which the optimal solution will be searched;
- Non-linear limits relative to result variables. If a certain variable should not fall within these limits it will be necessary to include an appropriate penalization of the objective function.

Using a suitable optimization algorithm of a multi-variable function in the presence of linear and non-linear constraints, an investigation among a large number (millions) of projects, obtained from the possible combinations of the independent variables, is carried out in order to identify the best project.



n the one-dimensional hypothesis, it is assumed to be able to identify, a combination of: mean diameter, rotation velocity and thermodynamic conditions capable of fulfilling the conservation equations.

A turbine stage is then formed by:

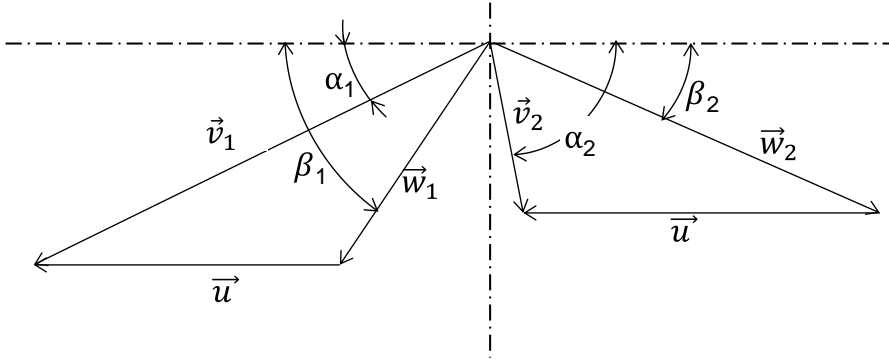
- A stator that has the task of expanding the fluid and accelerating it from velocity \vec{v}_0 (in the axial direction for the first stator, with a certain tangential component for any subsequent ones) to the velocity \vec{v}_1 in which the total enthalpy is conserved. The stator also diverts the flow giving it a tangential component. The velocity \vec{v}_1 is the absolute velocity at the rotor inlet.
- A rotor which has the task of deflecting the fluid and exchanging with it a certain momentum. The flow enters the rotor with a relative velocity \vec{w}_1 provided by the combination of the \vec{v}_1 vector and the \vec{u} vector representing the tangential velocity at the mean diameter of the stage:

$$u = \omega R = \omega \frac{D_m}{2} \quad (1.1)$$

The flow in the rotor can expand further (reaction stages) and $\vec{w}_2 > \vec{w}_1$ will result. Otherwise (impulse stages) a simple deflection of the flow will occur with \vec{w}_2 slightly lower to \vec{w}_1 , due to losses in the rotor rows.

The velocity \vec{w}_2 can then be composed with the \vec{u} vector to form \vec{v}_2 : absolute velocity at the exhaust stage.

Consistently with this description, it is possible to plot the velocity triangles, which represent in the two sections (1-1 and 2-2) the three \vec{v} vectors (absolute velocity), \vec{w} (relative velocity with respect to an observer integral with a system that rotates at the peripheral velocity \vec{u}), linked by the vector relation: $\vec{v} = \vec{w} + \vec{u}$



Velocity triangles and related sign conventions

The velocity triangles are a key tool in the design of turbomachinery, they contribute to determine the performance of a blade row and introduce limits to the deflection, the minimum values of the exhaust angles and the ratios between velocities. When the efficiencies of individual blade rows are known, η_s and η_r allow to reconstruct the thermodynamic quantities in the expansion process, calculate blade heights in considered sections and the Eulerian work.

Basic Equations

The fundamental equations are:

- mass conservation:

$$\dot{m} = \rho_0 \vec{v}_{0,a} \pi D_{m,0} h_0 = \rho_1 \vec{v}_{1,a} \pi D_{m,1} h_1 = \rho_2 \vec{v}_{2,a} \pi D_{m,2} h_2 \quad (1.2)$$

where \dot{m} is the effective mass flow rate of the turbine stage.

In the event of partial stages the degree of admission $GDA < 1$ must be considered, defined as a fraction of the frontal area through which the fluid flows.

$$\dot{m} = \rho \vec{v}_a \pi D_m h GDA \quad (1.3)$$

- energy conservation:

- between section 0-0 and 1-1 the total enthalpy h_t is conserved in the stator row

$$h_0 + \frac{\vec{v}_0^2}{2} = h_1 + \frac{\vec{v}_1^2}{2} \quad (1.4)$$

- between section 1-1 and 2-2 the total enthalpy $h_{t,r}$ is conserved in the rotor row

$$h_1 + \frac{\vec{w}_1^2}{2} = h_2 + \frac{\vec{w}_2^2}{2} \quad (1.5)$$

- between sections 0-0 and 2-2 it is possible to calculate the Eulerian stage work

$$h_0 + \frac{\vec{v}_0^2}{2} = h_2 + \frac{\vec{v}_2^2}{2} + L_{eu} \quad (1.6)$$

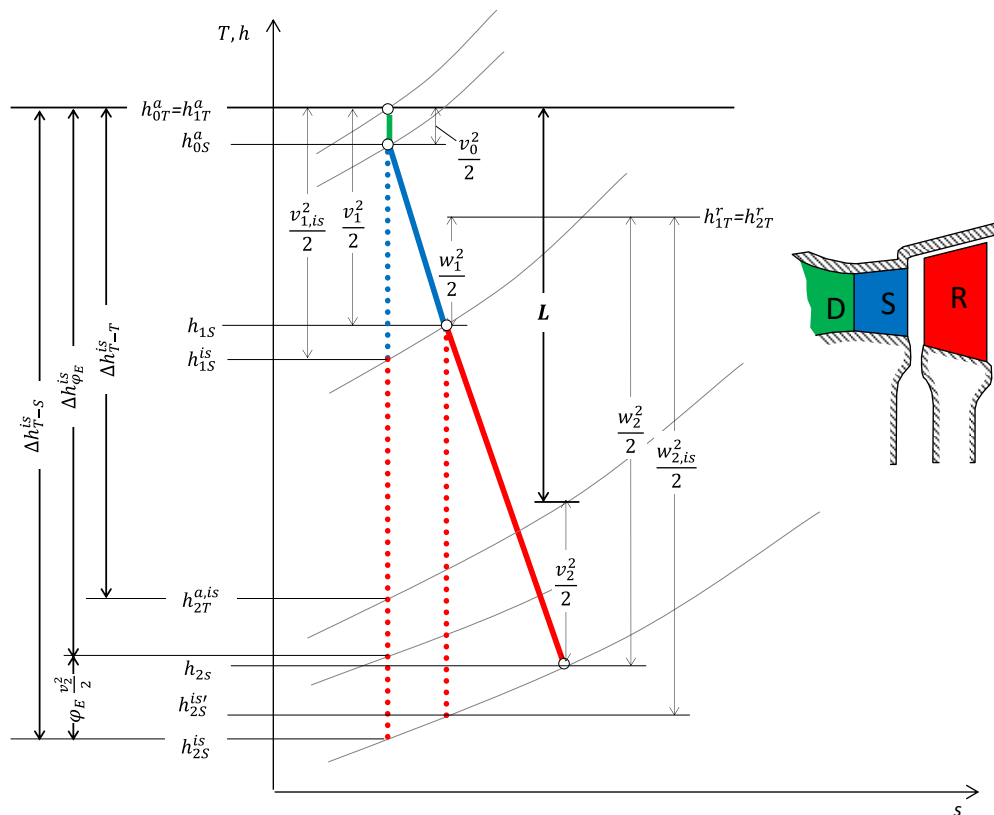
where L_{eu} is the work per unit of flowing mass transferred from the flow to the blades of the machine, it is also equal to:

$$L_{eu} = \vec{u}_1 \vec{v}_{1,t} - \vec{u}_2 \vec{v}_{2,t} \quad (1.7)$$

In real machines, there is sometimes a certain increase in the mean diameter of the rotor stage, implying an increase in the tangential velocity, and therefore $\vec{u}_2 > \vec{u}_1$. Very often it is possible to consider, as a first approximation, the perfect axial stage, and therefore

$$L_{eu} = \vec{u} \Delta \vec{v}_t \quad (1.8)$$

The figure below shows the expansion in a T-s or h-s diagram of one axial turbine stage in which, for simplicity, it is assumed that $\vec{u}_1 = \vec{u}_2$.



Temperature-entropy diagram of the expansion in an axial turbine stage (for simplicity of the representation it is assumed $u_1 = u_2$)

In the process of dimensioning an axial turbine stage, it is also useful to define the following dimensionless parameters:

- head coefficient of one stage related to real velocities

$$\psi = \frac{L_{eu}}{\vec{u}^2} = \frac{\vec{u} \Delta \vec{v}_t}{\vec{u}^2} = \frac{\Delta \vec{v}_t}{\vec{u}} \quad (1.9)$$

- isentropic head coefficient related to the total-to-static isentropic enthalpy change

$$k_{is} = \frac{\Delta h_{is T \rightarrow S}}{\frac{\bar{u}^2}{2}} \quad (1.10)$$

- flow coefficient which defines the axial component of the velocity vector at the exhaust of a blade row

$$\phi_1 = \frac{\vec{v}_{a,1}}{\bar{u}}, \quad \phi_2 = \frac{\vec{v}_{a,2}}{\bar{u}} \quad (1.11)$$

- degree of reaction which represents the fraction of the isentropic enthalpy change (linked to the expansion ratio) dissipated by the rotor compared with the isentropic total up to the static conditions

$$\chi = \frac{\Delta h_{rot,is}}{\Delta h_{is T \rightarrow S}} = 1 - \frac{\Delta h_{stat,is T \rightarrow S}}{\Delta h_{is T \rightarrow S}} \quad (1.12)$$

The degree of reaction can also be expressed referring to the isentropic velocity. Let us consider that the total isentropic enthalpy is given by the sum $\Delta h_{stat,is T \rightarrow S}$ necessary to accelerate the fluid up to $\vec{v}_{1,is}$ and of $\Delta h_{rot,is S \rightarrow S}$ necessary to accelerate the fluid from $\vec{w}_{1,is}$ to $\vec{w}_{2,is}$

$$\chi = \frac{\vec{w}_{2,is}^2 - \vec{w}_{1,is}^2}{\vec{v}_{1,is}^2 - \vec{w}_{2,is}^2 - \vec{w}_{1,is}^2} \quad (1.13)$$

The degree of reaction defines the type of stage that in the two most common cases take the name of the impulse stage ($\chi = 0$) and reaction stage ($\chi = 0.5$).

The parameters defined above are, of course, related to the shape of the velocity triangles, and the designer has a wide possibility of choices within certain limits defined by the loss ratios.

The efficiency of the stage is equal to the Eulerian work relative to an adequate isentropic enthalpy change which can be chosen considering the presence of a diffuser with a certain efficiency that slows down the current at the machine exhaust.

$$\eta = \frac{L_{eu}}{\Delta h_{is}} \quad (1.14)$$

- if there is no diffuser present, or if the diffuser has efficiency tending towards zero, all the kinetic energy at the exhaust is lost and one must refer to $\Delta h_{is\ T \rightarrow S}$ i.e. at the diffuser outlet there is a static pressure equal to that found downstream of the rotor. The efficiency is lowest among those considered.
- There is a diffuser with a certain performance φ_E that allows the recovery of a portion of the absolute kinetic energy at the rotor exhaust. The reference enthalpy change is equal to:

$$\Delta h_{is\ \varphi_E} = \Delta h_{is\ T \rightarrow S} + \varphi_E \frac{\vec{v}_2^2}{2} \quad (1.15)$$

- The diffuser has unitary efficiency and is able to fully recover the kinetic energy at the exhaust. In this case, one must refer to $\Delta h_{is\ T \rightarrow T}$. This hypothesis is adopted for intermediate stages, in which the energy at the exhaust is completely recovered by the next stage.

For each phase, it is possible to define the efficiency equal to the ratio between the kinetic energy at the exhaust and the one resulting from an isentropic process:

$$\eta_s = \frac{\vec{v}_1^2}{\vec{v}_{1,is}^2}; \quad \eta_r = \frac{\vec{w}_2^2}{\vec{w}_{2,is}^2} \quad (1.16)$$

where the kinetic energies are expressed with respect to the absolute velocity if the blade row is a stator or to the relative velocity if the blade row is a rotor.

In both cases, the expansion starts from the total upstream quantities (total relative, if the blade row is a rotor) and terminates at the static downstream pressure.

We have already seen how very often success in a machine design coincides with obtaining high efficiency. To this end, it is necessary to minimize the losses that occur in the machine, and therefore understand the connections between these and the various design variables. The baseline relation is:

$$\eta = 1 - \xi = 1 - \sum_i \xi_i \quad (1.17)$$

where the loss coefficient ξ depends on many parameters related to velocity triangles and the geometry of the machine. The search for optimal combinations of these parameters is subject to numerous constraints.

It is possible to categorize the limits to velocity triangles in:

- Relative limits: these refer to the "shape" of the velocity triangles, and therefore to angles and velocity ratios. It is necessary to analyse the influence of:
 - Exhaust angles α_1 and β_2
 - Angular deflection $\Delta\alpha$ and $\Delta\beta$
 - Ratio between relative velocities $|\vec{w}_2|/|\vec{w}_1|$
 - Exhaust angles α_2
- Absolute limits: these refer to the modules of vector velocities and the presence of supersonic flows. It is necessary to analyse the influence of:
 - Peripheral velocity \vec{u}_m for both the centrifugal and fluid-dynamical stresses

- Presence of supersonic flows at the exhaust and at the blade row inlet

Types of Stages in Axial-flow Turbine

The stages of a turbine can be characterized by a different distribution of the enthalpy change and different head coefficients.

The two extreme cases are the impulse stage and reaction stage at 50% that will be initially described as ideal (and then with the unitary efficiency of the blade rows) and then as real stages.

- The impulse stage has, by definition, the parameter $\chi = 0$: all the enthalpic drop is dissipated in the stator and the fluid arrives at the inlet of the rotor in a condition of static pressure equal to that found at the machine exhaust. The exhaust velocity from the $\vec{v}_{1,IS}$ stator is at its maximum and will be deflected by a small angle α_1 .

$$\vec{v}_{1,IS} = \sqrt{2 \Delta h_{IS T \rightarrow S}} \quad (1.18)$$

The rotor must only deflect the fluid to extract mechanical power and the relative velocity remains unchanged $\vec{w}_{1,IS} = \vec{w}_{2,IS}$ given that neither frictions nor changes in the mean diameter or blade height are present.

For the continuity relation, and considering that the frontal flow area remains constant, given that the relative velocities $\vec{v}_a = \vec{w}_a$ are symmetrical and $\beta_1 = \beta_2$ the Eulerian work stage is then:

$$L_{eu} = \vec{u} \Delta \vec{v}_t = \vec{u} [\vec{v}_{1,IS} \cos \alpha_1 + (\vec{v}_{1,IS} \cos \alpha_1 - \vec{u} - \vec{u})] = 2\vec{u} (\vec{v}_{1,IS} \cos \alpha_1 - \vec{u}) \quad (1.19)$$

The efficiency is the ratio between the Eulerian work and the isentropic enthalpy change:

$$\eta = \frac{L_{eu}}{\Delta h_{is\ T \rightarrow S}} = \frac{2\vec{u}(\vec{v}_{1,is}\cos\alpha_1 - \vec{u})}{\frac{\vec{v}_{1,is}^2}{2}} = 4\left(\frac{\vec{u}\cos\alpha_1}{\vec{v}_{1,is}} - \frac{\vec{u}^2}{\vec{v}_{1,is}^2}\right) \quad (1.20)$$

which vanishes for $\vec{u} = 0$ and for $\vec{u} = \vec{v}_{1,is}\cos\alpha_1$ and presents a maximum for $\vec{u} = \frac{\vec{v}_{1,is}\cos\alpha_1}{2}$

The maximum efficiency is therefore equal to:

$$\eta_{az} = \cos^2\alpha_1 \quad (1.21)$$

The only loss is represented by the kinetic energy at the exhaust and thus $\eta_{T \rightarrow T} = 1$.

Therefore, the smaller is the angle α_1 , the better is the theoretical efficiency. The real performance, as will be seen below, is instead strongly penalized at low values of α_1 , due to fluid-dynamic losses.

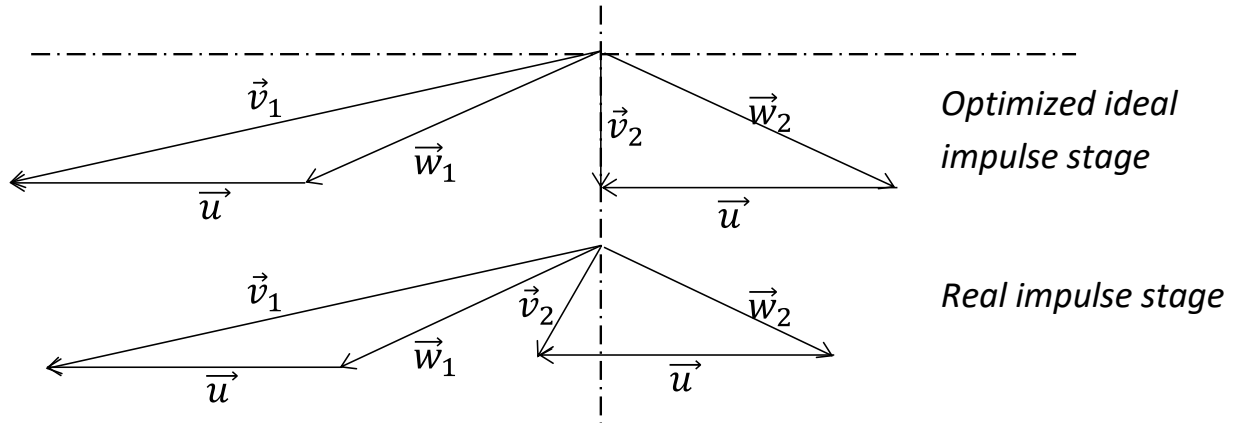
In maximum conditions the losses must be minimal; and in this case the loss of kinetic energy at the exhaust must therefore be minimized, which is in fact minimized by imposing an axial velocity $\vec{v}_{2,is}$.

From previous relations we find that the head coefficient is equal to:

$$k_{is} = \frac{4}{\cos^2\alpha_1} \quad (1.22)$$

and then, for small α_1 angles, with values of about 4.2 - 4.4.

In the case of a real stage, it is necessary to take into account the presence of an efficiency lower than 1 for both the stator and the rotor. It follows that $\vec{v}_1 < \vec{v}_{1,is}$, $\vec{w}_1 < \vec{w}_{1,is}$ and that $\vec{w}_2 < \vec{w}_1 < \vec{w}_{2,is}$.



Optimized velocity triangles for an ideal impulse stage and a real impulse stage

- The 50% reaction stage has, by definition, the parameter $\chi = 0.5$: of the total enthalpy change, half is dissipated in the stator and the absolute isentropic velocity in section 1-1 is lower than in the previous case.

$$\vec{v}_{1,is} = \sqrt{2(1-\chi)\Delta h_{is\ T\rightarrow S}} \quad (1.23)$$

The rotor further expands the fluid and the relative velocity at the exhaust will be higher than at the inlet $\vec{w}_{2,is} > \vec{w}_{1,is}$

$$\vec{w}_{2,is} = \vec{w}_{1,is} + \sqrt{2\chi\Delta h_{is\ T\rightarrow S}} \quad (1.24)$$

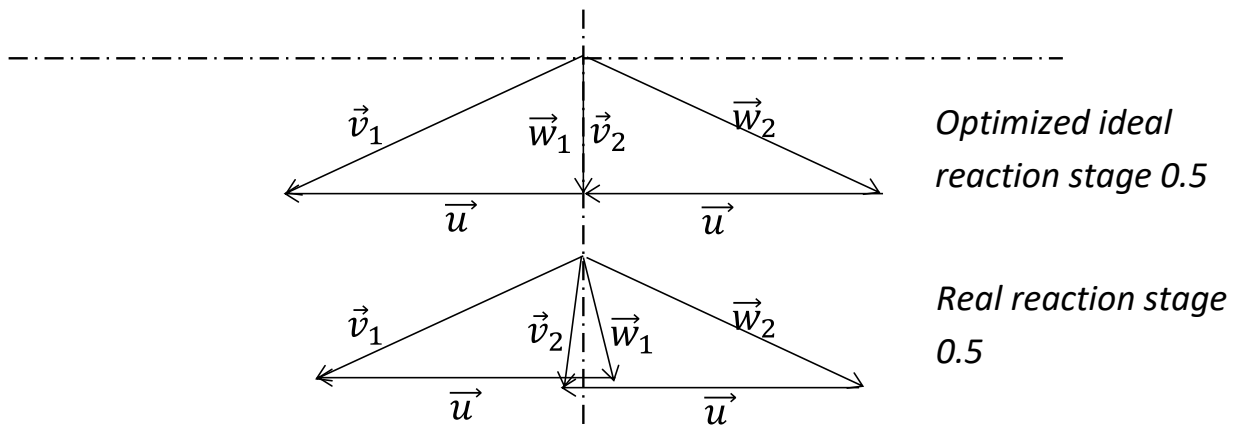
The efficiency, taking into account the definition of the degree of reaction referred to the velocities will be expressed as:

$$\eta = \frac{L_{eu}}{\Delta h_{is\ T\rightarrow S}} = 2\vec{u} \frac{(\vec{v}_{1,is}\cos\alpha_1 + \vec{v}_{2,is}\cos\alpha_2)}{\vec{v}_{1,is}^2 - \vec{w}_{2,is}^2 - \vec{w}_{1,is}^2} \quad (1.25)$$

The maximum efficiency is achieved for a value of the peripheral velocity equal to $\vec{u} = \vec{v}_{1,is}\cos\alpha_1$, and a degree of reaction equal to $\chi = 0.5$, and its value is higher than that of an impulse stage

$$\eta_{re} = \frac{\cos^2 \alpha_1}{1 - \cos^2 \alpha_1} \quad (1.26)$$

The head coefficient is lower and has values of about 2.2 - 2.4.



Optimized velocity triangle of an ideal reaction stage and a real reaction stage

A comparison between these ideal stages leads us to affirm that, given an identical peripheral velocity \vec{u} , it is preferable to adopt, when possible, a reaction stage because its efficiency limit is higher compared with an impulse stage and also the real efficiency is higher, as the stage is less affected by dissipative phenomena related to turbulent phenomena. On the other hand, a reaction stage is capable of dissipating a lower enthalpy change, and therefore the same total enthalpy change will require a higher number of stages. Impulse stages, on the contrary, have greater blade heights and are thus preferable when we have very small volumetric flow rates to increase the blade height and improve the efficiency of the stage.[2]

2. Axtur

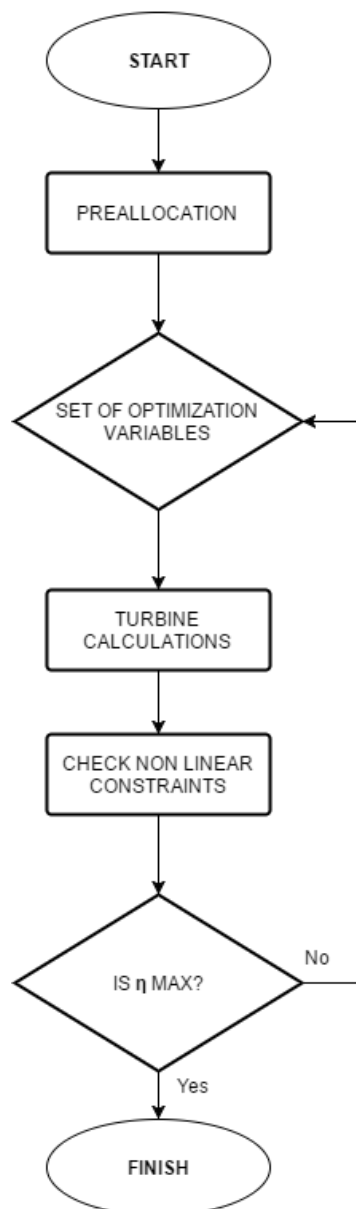
Axtur is a computer model developed to calculate and then optimize the geometry of an axial turbine aiming at the maximization of the overall efficiency. The first version of the program was “written” on punched cards by Professor Ennio Macchi in the sixties. Later converted in Fortran77 by Professor Giovanni Lozza the program was almost unchanged until nowadays. Based on a 1D model, Axtur designs the geometry of an axial turbine adopting many loss correlations available in the literature (e.g. Crag-Cox, Ainley-Mathieson, Traupel, etc...). For each stage of the turbine 9 variables are controlled by the optimization process to modify the geometry with the aim of maximizing the efficiency. Due to the limited computing power available in the seventies, some compromise had to be made in order to converge to a solution in a feasible amount of time. Ideal fluid hypothesis was adopted for the thermodynamic equations (although a pseudo-real setting was present, taking into account the compressibility factor of a fluid). The results achieved by the program were remarkable for the time. Today the FORTRAN version of Axtur runs an optimization process in less than a second on a consumer laptop or desktop machine.

In this thesis I'm going to analyse and discern the work that was done to convert the FORTRAN version of Axtur to MATLAB and the main features that were added to the original design.

The first feature that was changed from FORTRAN was the optimization algorithm, switching from the integrated one (Sbuzza) to the optimization tools available in MATLAB. This inherently solves another problem, the maximum number of stages, limited by the maximum number of variables that was able to handle the Sbuzza optimization algorithm. In this chapter various optimization algorithms are analysed and compared, while in the next chapter (3. New features) the effects of real gas equations and increment on number of stages are discerned.

2.1 Optimization process

The aim of the optimization process is to maximise the overall efficiency of a turbine operating on some geometrical parameters. The user can choose the starting parameters for the first calculation, and then the optimization function will try different design varying these parameters within their bounds until the best solution is found. At the end of each simulation the program also checks that non-linear constraints are not violated and in case adds small penalties to the final solution.



The objective function to be optimized is the overall efficiency of the turbine defined as:

$$\eta = \frac{W}{\Delta H_{T-S} - \varphi_E \frac{v_{2,a}^2}{2}}$$

where φ_E is the efficiency of the diffuser;

In the table are shown the variables controlled by the optimization function to try to maximise the efficiency. The kis is controlled only for the first stage, while the pressure ratio of the last stage is calculated knowing the overall pressure ratio for the machine and the pressure ratios of the previous stages.

First stage		Mid stages		Last stage	
cdis	kis	cdis	Diameter ratio	cdis	Diameter ratio
rzgd	Reaction extent	rzgd	Reaction extent	rzgd	Reaction extent
osx	Stator throat/pitch	osx	Stator throat/pitch	osx	Stator throat/pitch
osx	Rotor throat/pitch	osx	Rotor throat/pitch	osx	Rotor throat/pitch
ob	Stator throat/axial chord	ob	Stator throat/axial chord	ob	Stator throat/axial chord
ob	Rotor throat/axial chord	ob	Rotor throat/axial chord	ob	Rotor throat/axial chord
br	Stator axial chord/mean radius	br	Stator axial chord/mean radius	br	Stator axial chord/mean radius
br	Rotor axial chord/mean radius	br	Rotor axial chord/mean radius	br	Rotor axial chord/mean radius
rapp	Pressure ratio	rapp	Pressure ratio		

These parameters are the optimization variables of the problem and they can be varied by the optimization algorithm between a lower and an upper bound whose values are defined according to the limits of the correlations used to compute the efficiency losses. The limit on axial chord/mean radius is suggested by consideration on the rotor weight.

For the process to start, the starting values must lead to a feasible solution different from NaN (Not a number). The kis of the first stage is probably the most influential parameter for this matter. In case of bad initialization, $axtur$ try to adjust the value of the kis of the

first stage changing its value between the upper and the lower bound until a starting solution is found.

$$u = \sqrt{\frac{2\Delta H_{stage}}{k_{is}}}$$

$$d = \frac{u}{\omega}$$

Non-linear constraints on the other hand are checked at the end of each function evaluation. In the table are shown all the non-linear constraints in Axtur.

The h/d ratio is limited to 0.3 because over that limit three-dimensional effects due to variation of the peripheral speed along blade high takes place.

Mach number at rotor inlet must be limited to stay within the range of validity of the correlations and because supersonic inlet velocity on high deflection blades yield high losses. The limit on the outlet Mach is to guarantee converging channel in the rotor.

Limit on the flaring angle to avoid important radial velocity component in the flow that would fall outside of the validity of 1-D analysis.

The minimum ratio between h2 and h1 is limited to avoid convergence of the profile.

Non-linear constraints
Number of rotor blades
Number of stator blades
Rotor h/d ratio
Mach number at rotor inlet
Mach number at rotor outlet
Rotor h2/h1 ratio
Peripheral velocity
Stator hub flaring
Stator tip flaring
Rotor hub flaring
Rotor tip flaring
Stator throat
Rotor throat
Stator axial chord
Rotor axial chord
Rotor deceleration

The user in the starting parameters can define minimum and maximum values of non-linear constraints. If, at the end of each iteration, a constraint is not satisfied, the solution is penalized in order to lead the optimization process to discard such solution. The importance of every non-linear constraint can be defined by the user among the other parameters and is used by the program to choose the extent of the penalty to apply to the solution when the constraint is not satisfied.

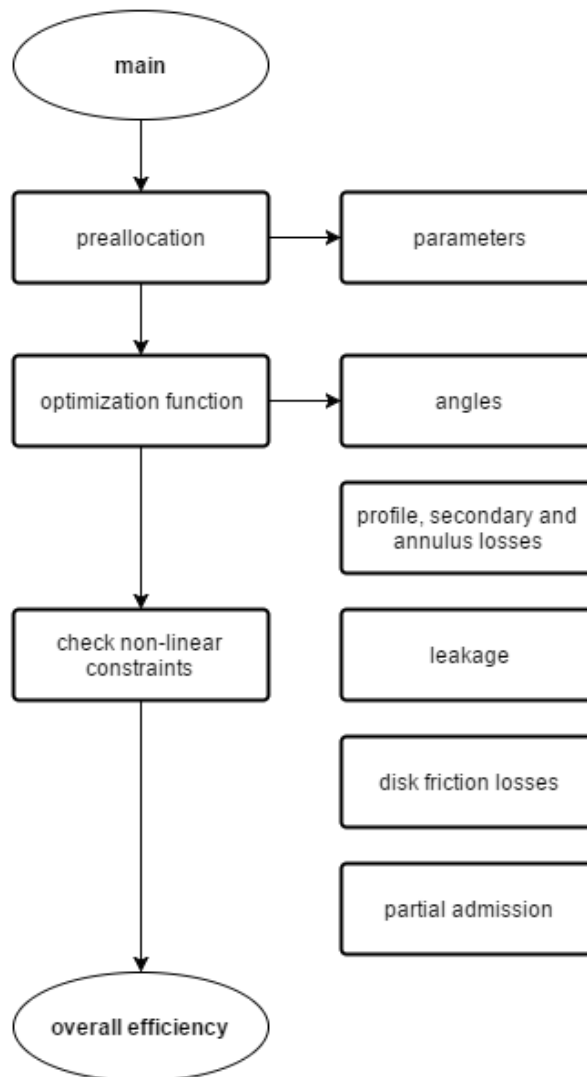
The choice of the optimization method is of key importance concerning quality of the solution and the amount of time needed for convergence. Three different functions are analysed in this section:

- **Fmincon:** The optimization problem is a nonlinear program (NLP) and it is tackled with gradient-based optimization algorithms, such as sequential quadratic programming (SQP) or interior point. Because of the nonconvexity of the objective function and constraints and the large number of variables, the optimization algorithm may return a local minima or even an infeasible solution, if the starting point is not good.
- **Pattern search:** Generate a sequence of iterates with nonincreasing objective function values where each iteration is either an optional so-called “search step” or a so-called “poll step.” In the search step, the objective function is evaluated at a finite number of solutions lying on a mesh to try to find one that yields a lower objective function value than the current solution. The search step is optional and not essential for the convergence guarantees. The poll step consists in the exploration of the neighboring mesh points around the current best solution. To guarantee the convergence on continuously differentiable functions, the set of poll directions must positively span the solution space.
- **Genetic algorithms (GAs):** The basic idea is to apply laws inspired by natural evolution, such as inheritance, mutation, selection, and crossover, to simulate the evolution of a population of solutions (called individuals). GAs are capable of quickly finding promising regions of the search space but may take a relatively long time to reach the optimal solution. When using evolutionary algorithms, attention should be paid to the selection of the algorithm parameters, such as number of individuals, probability of mutation, type of crossover, etc. The convergence rate and the accuracy of the returned solution strongly depend on these parameters. On the other hand, unfortunately, no general guidelines can be given because the optimal algorithm setup is problem specific (i.e., it depends on the number of optimization variables and on the features of the problem functions and constraints). Thereby, these parameters have to be tailored to each

optimization problem by performing several computational tests and analyzing the results. It is also important to note that, because the evolutionary algorithms have random operators, results are different at each run. Thus, to assess the accuracy of the results and the reliability of the algorithm, the optimization should be repeated several times (at least 5e10 times), keeping the best solutions.[3]

Because of the complexity of genetic algorithms only fmincon and pattern search are analysis in this paper.

2.2 Algorithm and flowchart



Above is shown the simplified flowchart followed by the program for the solution of the algorithm that fully characterizes the geometry of the turbine. Starting from the main function a preallocation process prepares the variable needed by the program to perform its calculation. This include the parameters stored in the file "parameters.m" that, as the name suggests, are the input data that defines the problem. This process take less than a second if ideal gas hypothesis is chosen, while can take a few seconds in case of real gas calculation. Following, the core function of the program is called: funzot. This function is divided in five sections:

- Loading of optimization variables: the values of the independent variables, stored in an array managed by the optimizator, are assigned to their relative variable within the program.
- Stator: the stator geometry and velocity triangles are calculated. This requires two calls to the `angol` and `cox_ls` functions. The first taking in to account the post expansion phenomena after the blades, the latter calculating the various losses within the stator according to the Craig-Cox correlations.
- Rotor: Analogously to the stator, the rotor geometry is calculated.
- Losses: In this section, the losses due to leakage, disk friction and partial admission are calculated.
- Overall efficiency: once all the geometric values and losses are taken in to account, the overall efficiency is calculated.

The following effects are taken in to account in Axtur to characterize all the losses occurring in the turbine cascades:

- Profile loss (Craig and Cox, 1971): due to the blade shape and effects related to friction, fluid vane deflection, and boundary layer dissipation. This loss mainly depends on blade pitch, blade axial chord length, trailing edge thickness and roughness of blade surface, angular deflection, and relative velocity ratio.
- Secondary losses (Craig and Cox, 1971): caused by secondary flow structures mainly described by passage vortex, horseshoe vortex, trailing edge vortex, and

corner vortex. These losses are affected by the same parameters which have influence on profile losses plus blade height.

- Annulus losses (Craig and Cox, 1971), or Kacker and Okapuu (1981): due to the passage of fluid in the gap between two blade rows. This loss is calculated for all the stages except for the last one.
- Leakage losses (Craig and Cox, 1971): caused by the unwanted passage of fluid above the blade tip whose expansion does not contribute to power production. This kind of loss is mainly related to radial clearance, blade length, and blade overlap. They can be null for the first stator blades only.
- Disk windage losses: due to the velocity gradient in the clearance between stator and rotor disk walls. They are strongly affected by rotational speed and by both absolute disk clearance and disk diameter.
- Kinetic energy loss: it is the fraction of kinetic energy of discharge velocity which cannot be recovered with the diffuser. Usually it is defined with a coefficient smaller than unit respect to the kinetic energy of the axial component.[1]

A “vincol” function is then called to check whether or not non-linear constraints have been reached and if so a warning message is displayed in the console. To end the process “genout.m” generates a txt with the main parameters and four charts to better visualize the results of the process.

Here are shown, as an example, the output plots of a three stages turbine (after optimization) with the following starting parameters:

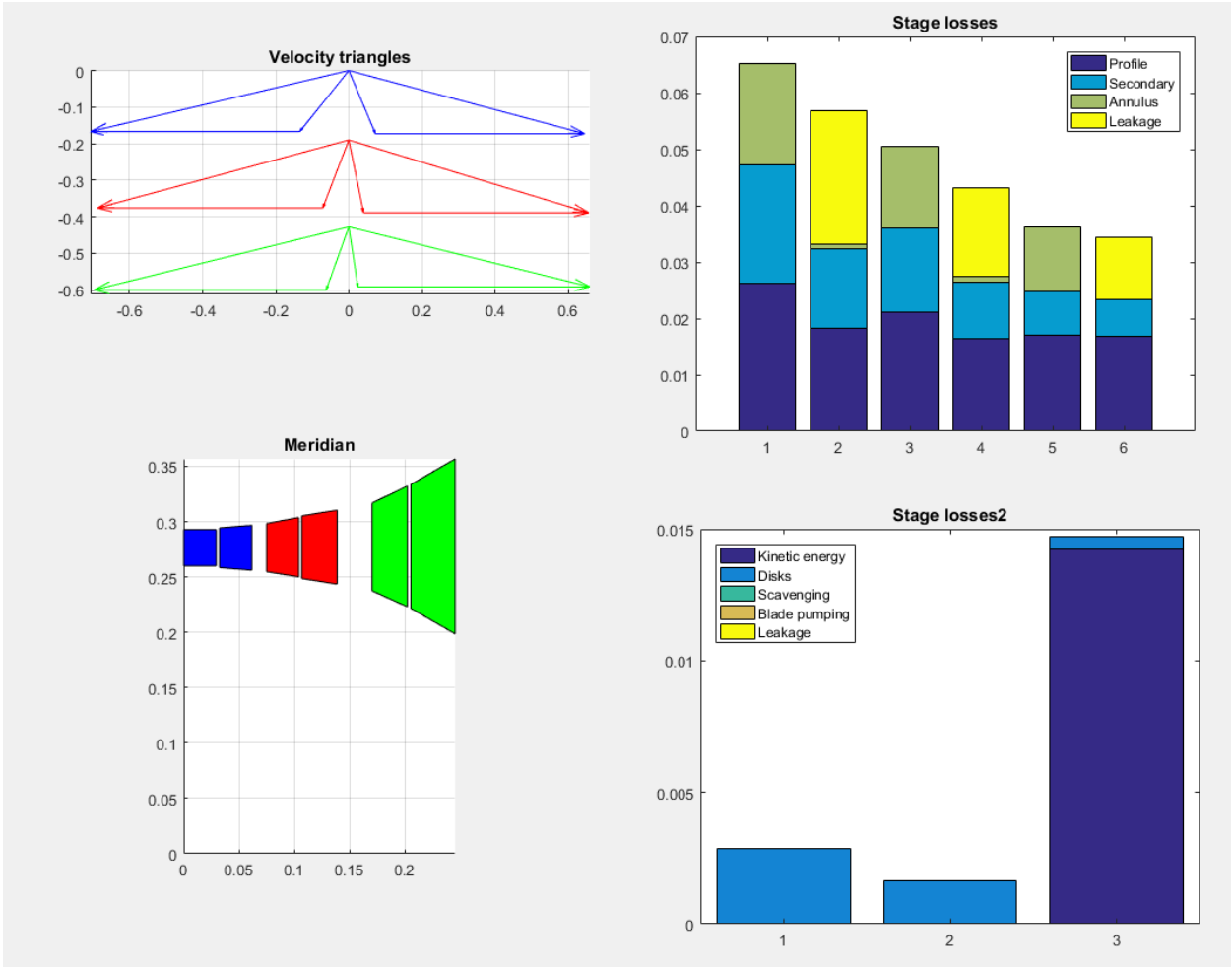
T_{in} : 650 K

P_{in} : 1 bar

Expansion ratio: 10

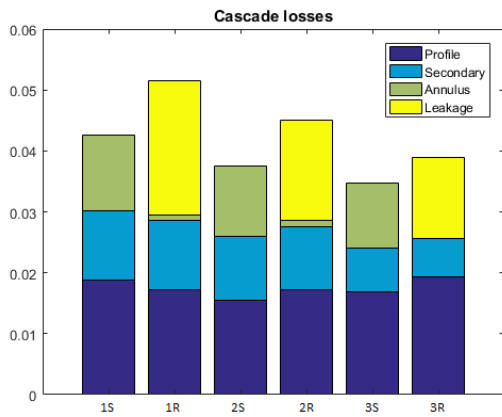
Mass flow rate: 2 kg/s

Fluid: Nitrogen (N_2)

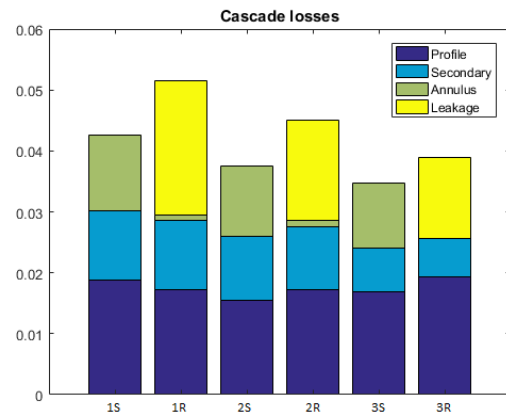


2.3 MATLAB Axtur vs. FORTRAN Axtur

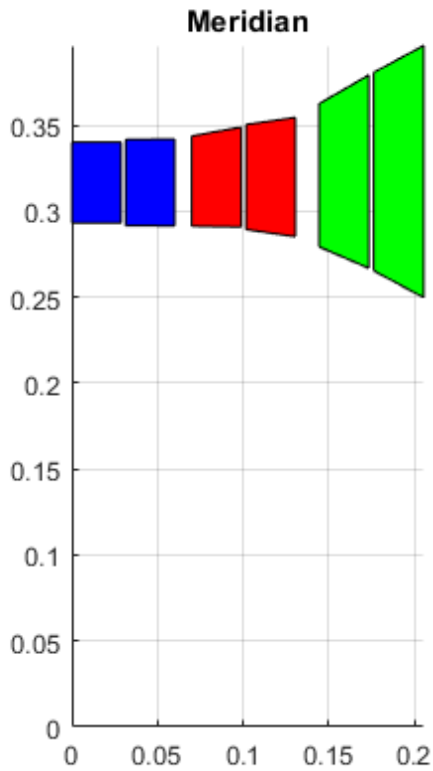
To double check that the conversion from FORTRAN to MATLAB was carried out in a correct way, a comparison of the result achieved by the two programs is analysed. Because of the difference of the optimization process the results from an optimization run on FORTRAN are fed as input for a single run on MATLAB. Ideal gas hypothesis was set to emulate the constant specific heat ratios adopted by the original Axtur. As expected, the results are the same, the meridian plane and the cascade losses shows the same shape and the numerical results have no differences.



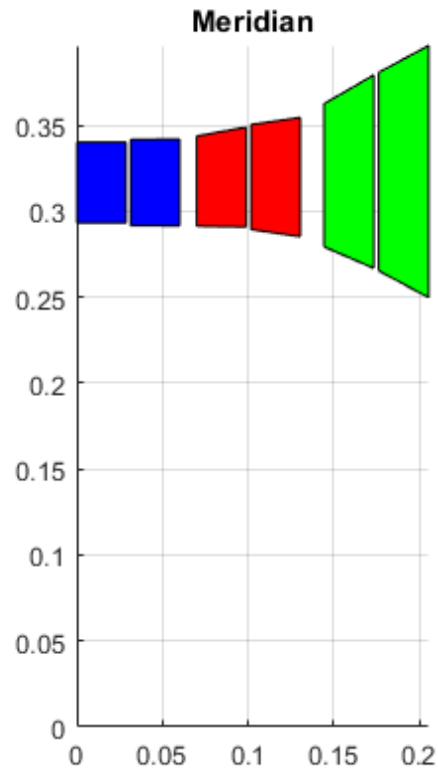
FORTRAN



MATLAB



FORTRAN



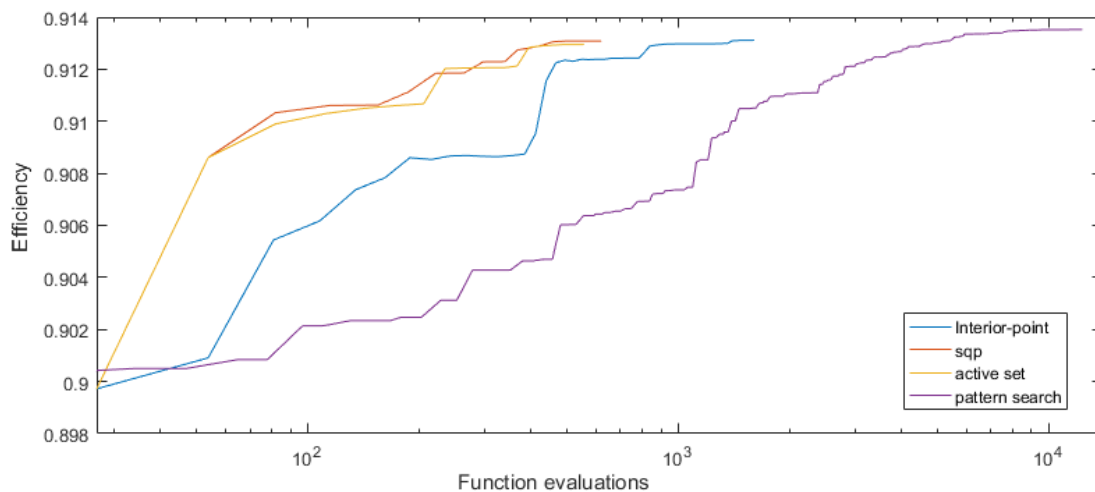
MATLAB

Fluid: Nitrogen, $T_{in} = 1123.15$ K, $P_{in} = 5$ bar, $\beta = 10$, mass flow rate = 10 kg/s

2.3 Analysis of optimization functions and algorithms

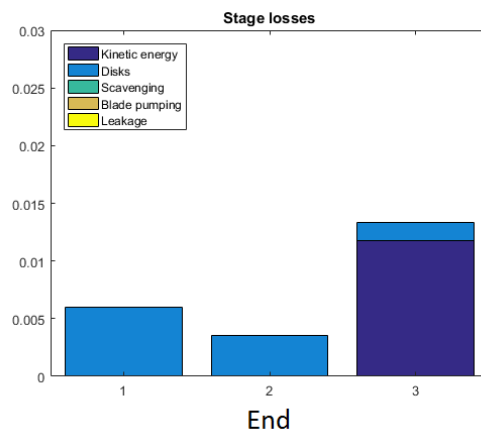
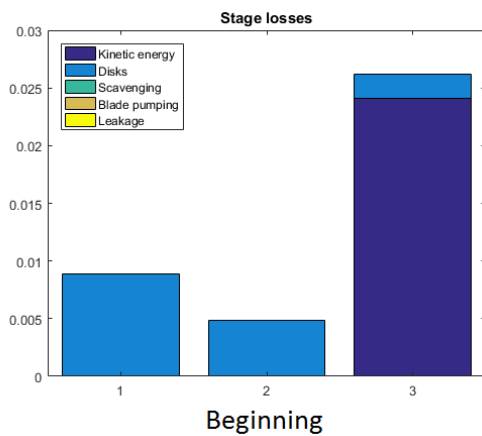
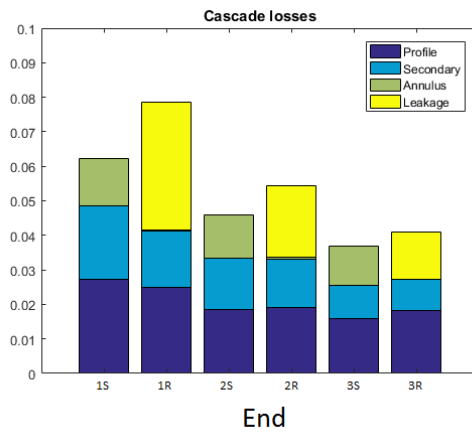
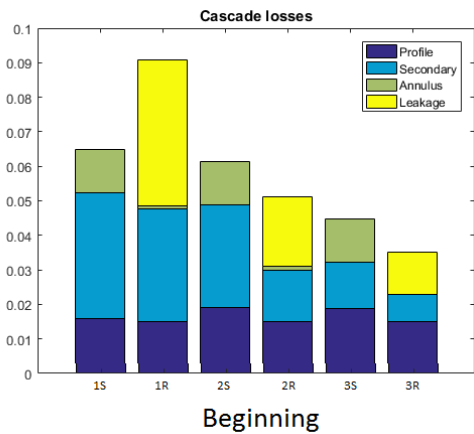
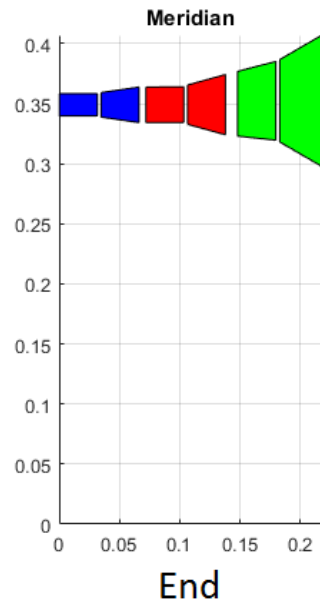
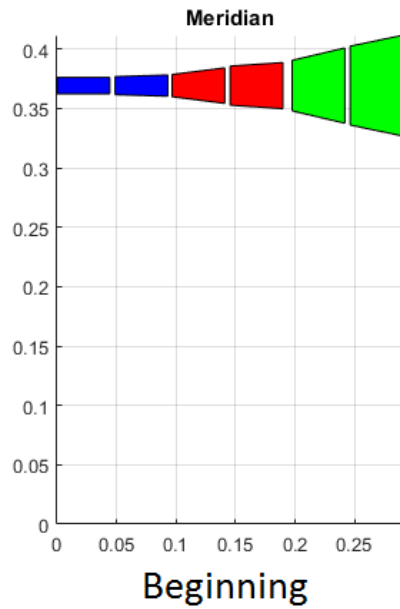
A comparison of the time and the quality of the optimization process is carried out to understand the differences between the various tools implemented in MATLAB. The analysis is performed on the three algorithms of the fmincon function (interior point, sqp and active set) as well as the pattern search.

The same starting parameters are fed to the four functions and the overall efficiency achieved is plotted versus the number of iteration needed to converge. On the x axis a logarithmic scale is adopted to allow the pattern search to be displayed on the same chart of the three algorithms of the fmincon.



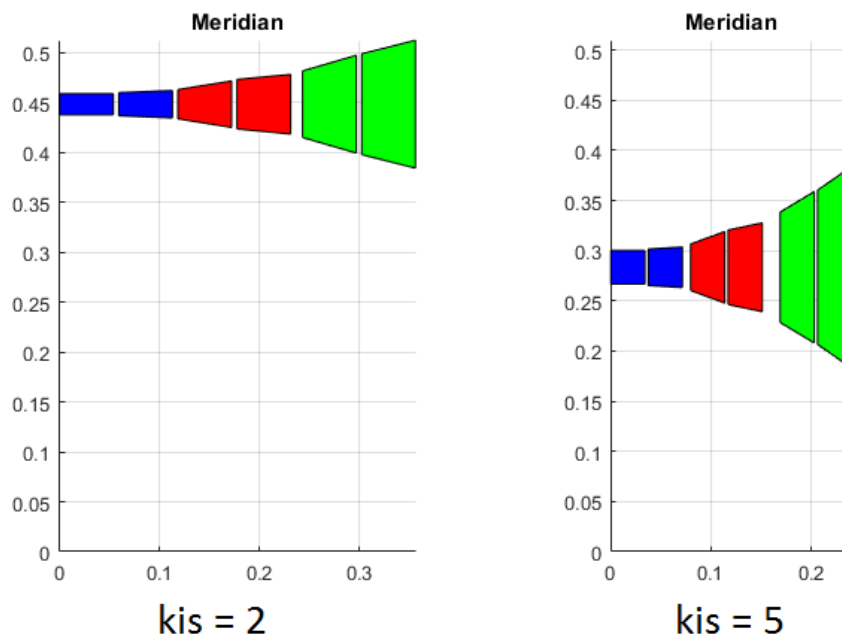
The results are very similar in terms of final overall efficiency but the number of iteration needed is very different, namely the fmincon function reach convergence much faster than the pattern search. Among the three algorithms of the fmincon the interior point is the slower, but still manages to reach the solution within a thousand evaluations. On the other hand, the pattern search needs 10 to 50 thousand iterations showing a very slow trend that lead to the solution. Because of the strong assumption of independence between all the losses and the very small difference in solution achieved by the pattern search, the fmincon is choose as the best optimization process for this problem.

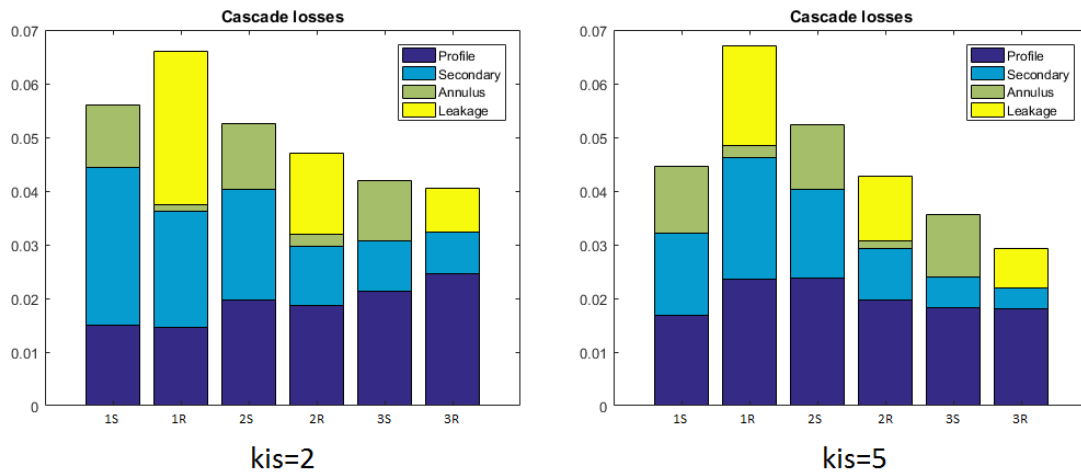
To understand the process carried out by the optimization, the starting and ending design and losses are plotted below.



The process starts from an already high efficiency of 89.9% and ends at 91.3%. As is shown from the meridian plane the k_{is} (and consequently the mean diameter) are already close to optimum and varies from the initial value of 3.00 to the final of 3.09 resulting in higher blades and consequently lower secondary losses. The degree of reaction is lower in the first stage, resulting in higher deflection of the fluid and higher profile losses. To reduce the kinetic energy to the exhaust the outlet velocity is almost parallel to the axial direction and the blade height at the exhaust is increased of about 20% growing from 0.08m to 0.10m.

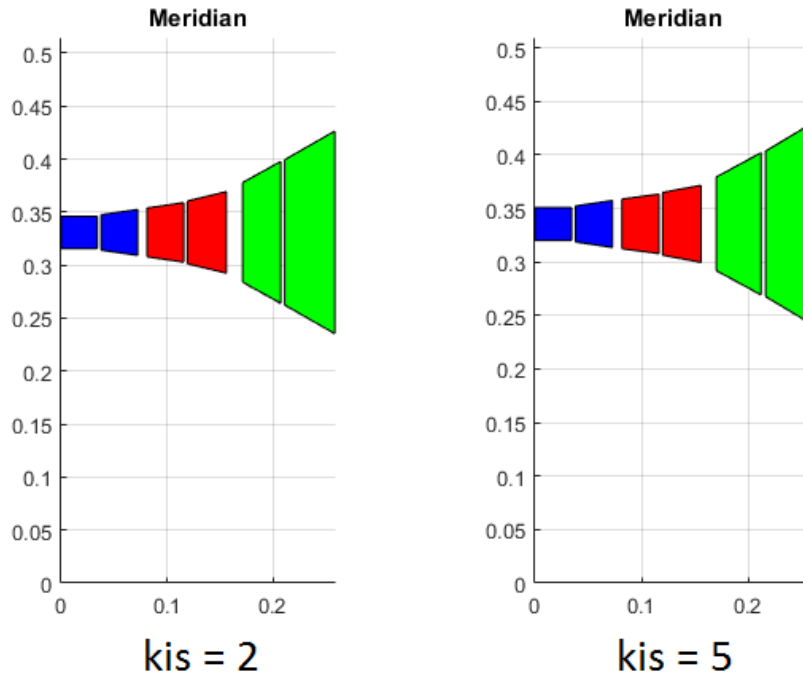
To further test the optimization algorithms two turbine with very different starting design are optimized to find out whether they converge to the same solution or not. A first case with $k_{is}=2$ showing a very high diameter and a second with a $k_{is}=5$ and a consequent low mean diameter.

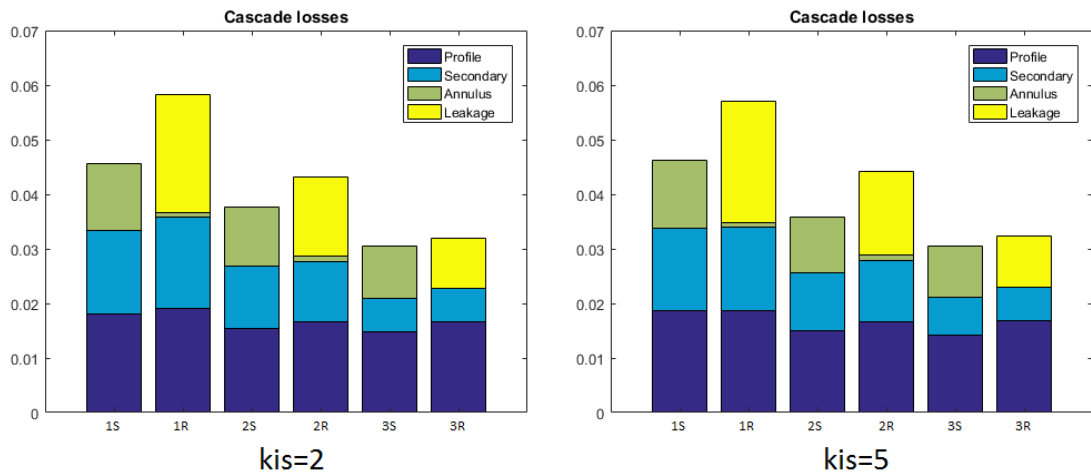




In the first, very high secondary losses are present due to the very low blade high, while in the second, due to a higher h/d , there is a reduction of these losses. Nevertheless the blades profiles are not optimized and shows high profile and secondary losses even in the $kis=5$ case.

The results after optimization are shown below.





Stage		$kis=2$	$kis=5$
1	Beta	2.4091	2.2671
	Kis	3.30681	3.027
	r^*	0.39364	0.39168
	h/d stator	0.0459	0.0456
	h/d rotor	0.06562	0.06559
	mach stator	0.81093	0.78344
	mach rotor	0.75729	0.71798
2	beta	2.7239	2.9095
	Kis	2.73586	2.84919
	r^*	0.45275	0.46208
	h/d stator	0.08486	0.08283
	h/d rotor	0.1163	0.10719
	mach stator	0.81348	0.83073
	mach rotor	0.85306	0.89307
3	beta	3.3016	3.2856
	Kis	2.22688	2.1511
	r^*	0.46966	0.48869
	h/d stator	0.20247	0.19798
	h/d rotor	0.28878	0.27904
	mach stator	0.86992	0.84865
	mach rotor	0.95438	0.96921
Overall efficiency		94.43%	94.45%

Although the overall efficiency (which is the objective function to be maximised) are almost the same, the combinations of variable that bring to that solution can be different as is shown in this case.

The turbine starting from a $k_{is}=2$ finds an optimum with a $k_{is}=3.3$ with an expansion rate of 2.41, while the second, with a $k_{is}=3.03$ and an expansion rate of 2.27. A higher beta means a higher volume flow rate that is discharged with a higher mach number leading to higher secondary losses in the first stage. This is compensated in the last stage, where a slightly higher h/d allows lower secondary losses and a lower mach number.

If run with the pattern search the two starting design converges to a much closer solution but taking about 10 times longer. Because this 1D simulation is an approximation of the real behaviour of the fluid and the strong assumption made on the independence of the various losses introduces a degree of uncertainty to the solution, the two solutions can be considered very similar. A further 2D and 3D analysis could, starting from this design, refine the geometry of the machine to allow a higher and more accurate efficiency prediction.

3. New features

In this section a detailed analysis of the new features introduced in the MATLAB version of Axtur is carried out. Beside the fact that the conversion to MATLAB itself is already an improvement, because its code doesn't need to be compiled in an executable file like FORTRAN, two main features were implemented after the initial conversion: Real gas simulation with the integration of REFPROP and the removal of the limit on the maximum number of stages. The first one is probably the most important and will be analysed in detail among different fluids and conditions.

3.1 Ideal vs. real gas hypothesis

For an ideal gas, a plot of PV/nRT versus P gives a horizontal line with an intercept of 1 on the PV/nRT axis. Real gases, however, show significant deviations from the behaviour expected for an ideal gas, particularly at high pressures. Only at relatively low pressures (less than 1 atm) do real gases approximate ideal gas. Under these conditions, the two basic assumptions behind the ideal gas law—namely, that gas molecules have negligible volume and that intermolecular interactions are negligible—are no longer valid.

Because the molecules of an ideal gas are assumed to have zero volume, the volume available to them for motion is always the same as the volume of the container. In contrast, the molecules of a real gas have small but measurable volumes. At low pressures, the gaseous molecules are relatively far apart, but as the pressure of the gas increases, the intermolecular distances become smaller and smaller. As a result, the volume occupied by the molecules becomes significant compared with the volume of the container. Consequently, the total volume occupied by the gas is greater than the volume predicted by the ideal gas law. Thus at very high pressures, the experimentally measured value of PV/nRT is greater than the value predicted by the ideal gas law.

A real gas hypothesis was implemented in the original Axtur that introduced a compressibility factor to the ideal gas law to simulate the real behaviour of fluids:

$$PV = ZnRT$$

The compressibility factor was defined at an initial point, at mid pressure and at outlet conditions and was linearly interpolated.

When running in real mode the user had to provide some quantities that were otherwise calculated in the ideal gas hypothesis. The overall enthalpy drop for the isentropic process, the polytropic expansion coefficient dH/dT for each stage, and the specific heat ratio calculated with p and v .

To improve the accuracy of the simulation when dealing with real gas and to make the program more user friendly, an external program called REFPROP was integrated to Axtur.

REFPROP is a program that uses equations for the thermodynamic and transport properties to calculate the state points of the fluid or mixture. These equations are the most accurate equations available worldwide. Their high accuracy is obtained through many coefficients in the equations, and thus the calculation speed will be slower than other equations such as Peng-Robinson cubic equations. The equations are generally valid over the entire vapour and liquid regions of the fluid, including super-critical states; the upper temperature limit is usually near the point of decomposition of the fluid, and the upper pressure (or density) limit is defined by the melting line of the substance.

The MATLAB version of Axtur can be run in two different modes: ideal gas or real gas.

- Ideal gas: Much similarly to the FORTRAN version of Axtur the specific heat ratios and consequently all the properties are kept constant through each iteration.
- Real gas: All the thermodynamic data is taken from REFPROP for each iteration, all the equations valid for ideal fluid are replaced by real process obtained by REFPROP.

Because each call to REFPROP takes same time, calling it each iteration would have make the program way too slow. To overcome this problem all the thermodynamic data needed by the program are preallocated during a preallocation process prior to the optimization. A “realgas.m” class was added to Axtur to fulfil this need. For each property two functions are defined: “set” and “get”, the former creates a table (100x100 by default) of values of the property and is called during preallocation, the latter linearly interpolates the table to get a value and is called within the optimization function.

Reported is the code from “realgas.m” regarding the enthalpy as function of pressure and temperature.

```

%% ENTHALPY
% Creates a p-t table of enthalpy values
function [htptab,htppar]=htp_set(pin,raptot,tin,fluid)
    dim    = 50;
    dim    = dim+1;
    s      = refpropm('s','t',tin,'p',pin/1000,fluid);
    tiso   = refpropm('t','p',pin/raptot/1000,'s',s,fluid)-1;
    rapt   = tin/tiso;
    htptab = zeros(dim,dim);
    a = fliplr(tin/rapt : tin*(1-1/rapt)/(dim-1) : tin);
    b = fliplr(pin/raptot : pin*(1-1/raptot)/(dim-1) : pin);
    for i=1:dim
        for j=1:dim
            try
                htptab(i,j) = refpropm('H','T',a(j),'P',b(i)/1000,fluid);
            catch
                htptab(i,j) = nan;
            end
        end
    end
    end
    htppar(1) = pin;
    htppar(2) = raptot;
    htppar(3) = tin;
    htppar(4) = rapt;
end

```

pin = total inlet pressure, tin = total inlet pressure, raptot = total expansion ratio.

As already anticipated in the introduction, the preallocation process takes some time for the real gas. Eleven properties have to be preallocated in matrixes of 10000 values each.

A total of 110,000 calls have to be made to REFPROP. This takes between 30 and 60 seconds on a 2011 desktop CPU rated at 83 GFLOPS.

Listed below are the properties needed by Axtur when the real gas hypothesis is chosen.

Property	Function of		Symbol	Unit
Specific heat ratio	Temperature	Pressure	$\gamma(T,P)$	-
Compressibility factor	Temperature	Pressure	$z(T,P)$	-
Enthalpy	Temperature	Pressure	$h(T,P)$	J/kg
Speed of sound	Temperature	Pressure	$a(T,P)$	m/s
Density	Temperature	Pressure	$\rho(T,P)$	kg/m ³
Entropy	Temperature	Pressure	$s(T,P)$	J/(kg K)
Critical flow factor	Temperature	Pressure	$C^*(T,P)$	-
Temperature	Pressure	Entropy	$T(P,s)$	K
Temperature	Enthalpy	Entropy	$T(h,s)$	K
Pressure	Enthalpy	Entropy	$P(h,s)$	kPa
Throat mass flux	Temperature	Pressure	$q(T,P)$	kg/(m ² s)

To avoid making the “funzot.m” script too cumbersome filling it with IF statement to switch from ideal to real equations, a “funzot_real.m” is adopted instead when the real gas hypothesis is set in the parameters. Thanks to REFPROP providing the real value for the fluid in any condition, the properties are not kept constant throughout the whole stage like in the ideal hypothesis but instead their exact value is obtained in each point within the stage. As an example the same calculations from “funzot.m” and “funzot_real.m” are reported in the following page. They show the process performed at the beginning of the rotor where, from the static condition at stator outlet, the total conditions are derived.

```

%% ROTOR
    % section e1 (total inlet)
    % section 2 (rotor outlet) (corr. to 0 and 1 for
stator)
    u(ir) = u(is)*rr(i);
    te1 = t1+win(ir)^2/2/cx(i);
    pe1(i) = p1(i)*(te1/t1)^(1/theta(i));
    t2is = te1*(p2(i)/pe1(i))^theta(i);
    w2is = sqrt(2*cx(i)*(te1-t2is));
    % mach2, angles and throats
    am2is = w2is/sqrt(gampv(i)*rg*t2is);

```

From “funzot.m”.

The total inlet temperature “te1” is calculated by adding the kinetic energy at the stator outlet to the static temperature in the same section. It must be noted that the specific heat “cx” is calculated at the beginning of the stator and do not takes in to account the variation occurred within the stator blades. The total pressure “pe1” is calculated with “theta” that, much the same as “cx”, is kept constant throughout the whole stage. The isentropic velocity “w2is” and the speed of sound (calculated as $a = \sqrt{\gamma RT}$) are affected by the same issues.

```

%% ROTOR
    % section e1 (total inlet)
    % section 2 (rotor outlet) (corr. to 0 and 1 for stator)
    u(ir) = u(is)*rr(i);
    he1 = realgas.htp(t1,p1(i))+win(ir)^2/2;
    s = realgas.stp(t1,p1(i));
    te1 = realgas.ths(he1,s);
    pe1(i) = realgas.phs(he1,s);
    t2is = realgas.tps(p2(i),s);
    w2is = sqrt(2*(realgas.htp(te1,pe1(i))-
realgas.htp(t2is,p2(i))));
    % mach2, angles and throats
    am2is = w2is/realgas.atp(t2is,p2(i));

```

From “funzot_real.m”.

In order to get the total temperature value “te1” the total enthalpy “he1” is first calculated adding to the real value of the static enthalpy at stator outlet the kinetic

energy in the same section. (`realgas.htp(t1,p1(i))`) calls the linear interpolation of the enthalpy table, function of temperature and pressure, at the static condition in section 1). Analogously the entropy is calculated in the same section and the total temperature is defined as the temperature the fluid would have if taken to an isentropic stop. The total pressure is defined in the same way and the isentropic velocity is calculated with following equation:

$$w_{2is} = \sqrt{2(h_{e1} - h_2)}$$

The speed of sound, as the other properties, assumes its exact value at rotor outlet condition.

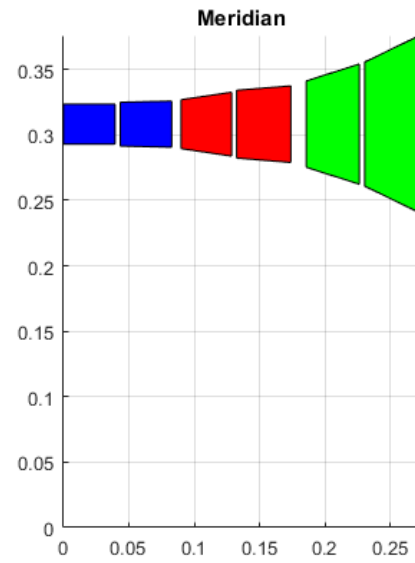
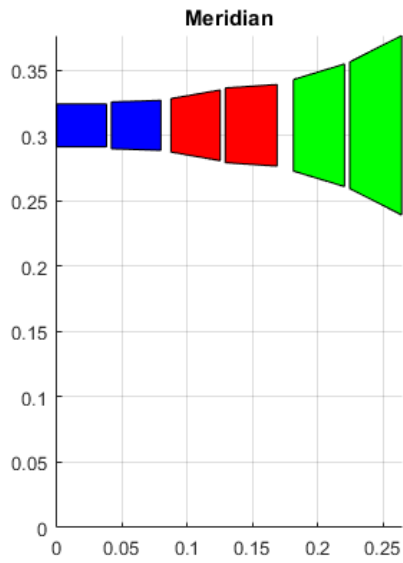
In the “angol.m” script are calculated the cinematic and mechanical angles at the outlet of the cascade and the deviation effect of the post expansion. Here are reported the calculation for a convergent-divergent nozzle for a real fluid.

For the ideal gas the critical throat section is calculated with the aid of the critical beta, defined from the specific heat ratio. The post expansion process is derived from an iterative process that, starting from the outlet conditions, reduces the Mach number until the condition at the cascade outlet section, defined by the parameter “ff” are met. In the end the deviation introduced by the post expansion is calculated according to the Vavra correlation.

In the real case, on the other hand, the critical throat is calculated by REFPROP and the iterative process move backward from the total outlet condition reducing the isentropic velocity until the condition defined by the same “ff” parameter are achieved. The deviation in this case is calculated according to the mass conservation and takes in to account the increase in the flowing section at the cascade outlet.

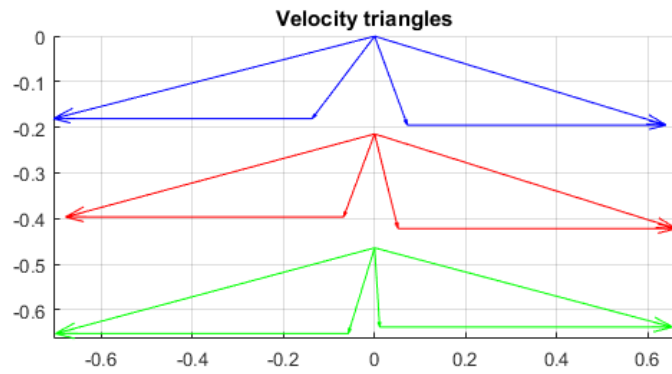
An analysis of the differences running the program with ideal and real hypothesis is carried out for three fluids: a noble gas (Neon), a diatomic gas (Nitrogen) and an hydrocarbon for ORC applications (Octane).

3.1.1 Neon

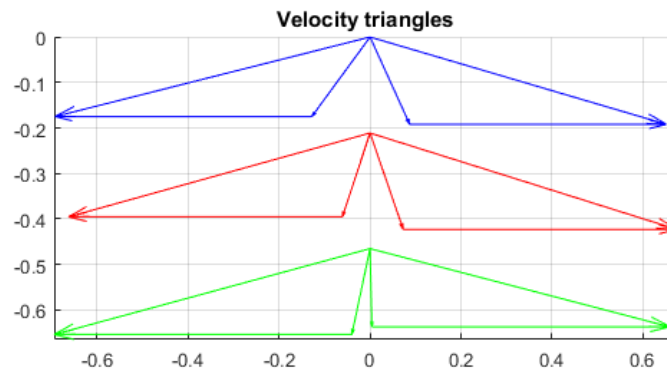


Ideal gas

Real gas



Ideal gas

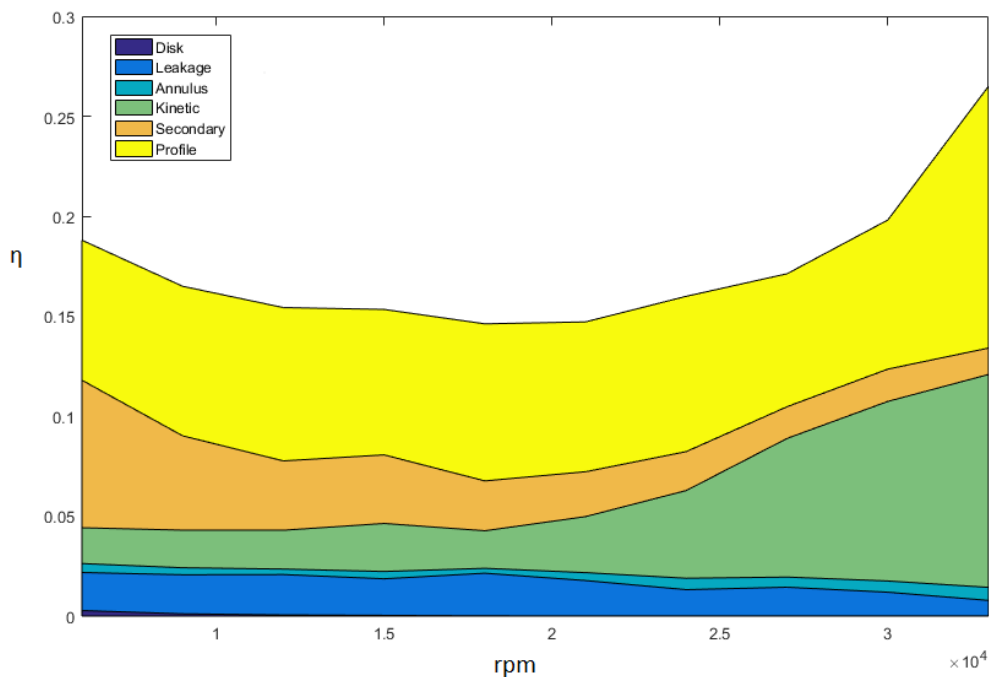


Real gas

Neon and noble gasses in general, are the real fluids that better approximate the behaviour of a real one due to their monoatomic molecules. They show a nearly constant specific heat ratio of about 1.66 that is very slightly influenced by temperature. Because of this reason, as it is clear from the plots, the differences in design between ideal and real fluids are almost inexistent.

The overall efficiency differs of less than 0.1% and the efficiencies, both total-to-static and total-to-total, of each stage show a decreasing differences as the pressure decrease and the fluid approach the ideal fluid hypothesis conditions.

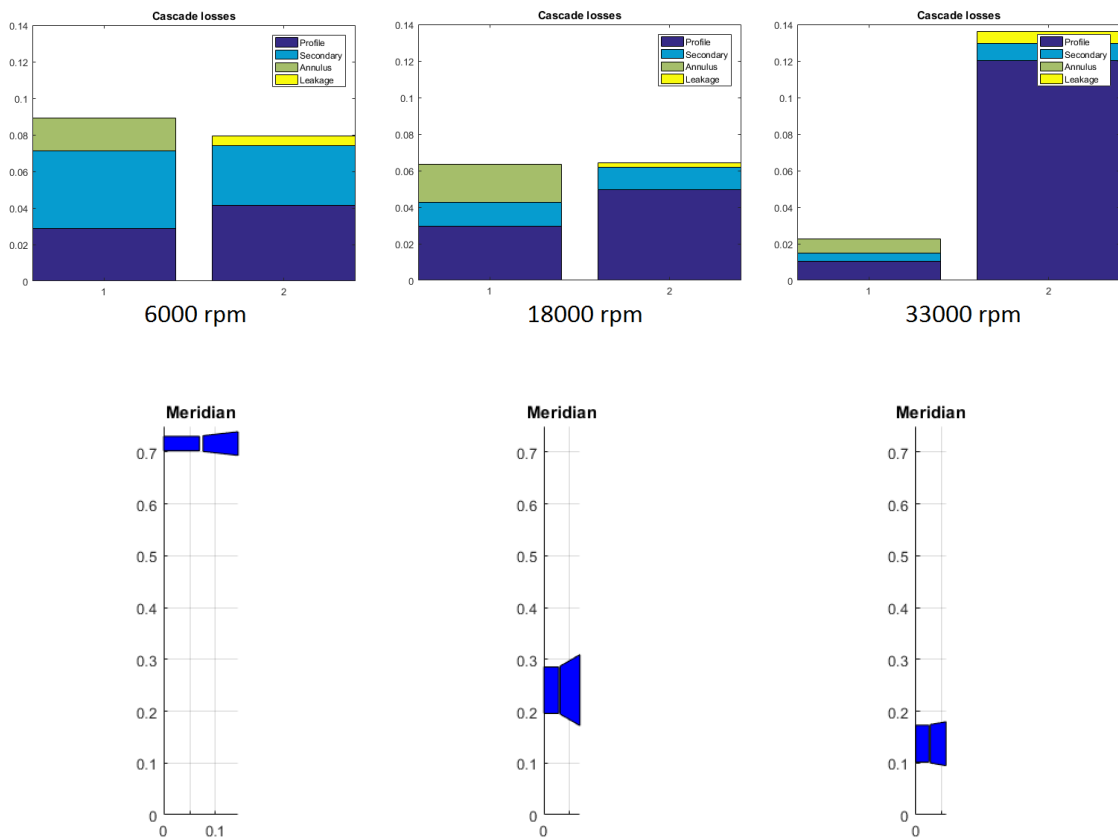
A single stage Neon turbine is take in consideration to show the behaviour of the various losses changing the rotation speed from the optimal conditions.



In the above chart are reported the cumulative effects of the losses taking place in the turbine cascades. The maximum efficiency is found around 18000 rpm where a good compromise between all the losses can be found. As the rotating speed decrease from the optimal point the secondary losses start to increase. This is due to their strict relation with the blade height. As it can be seen from the bottom plots of the meridian planes the low speed turbine have a large mean diameter needed to maintain the peripheral

velocity to an optimal point and reduce stage loading. Due to the small h/d both secondary and leakage losses increase, with the secondary strongly penalizing the overall efficiency.

On the other end of the chart a very high speed turbine shows large kinetic and profile losses due to the small diameter and the large exhaust velocity. The exhaust velocity has a large tangential component that cannot be recovered from the diffuser and causes a strong distortion in the velocity triangles that affects the profile losses.



Once the importance of rotational speed optimization is assessed a multi stage turbine adopting Nitrogen as a working fluid is analysed.

3.1.2 Nitrogen

Due to its diatomic molecule Nitrogen shows a specific heat that is not constant as the vibrational contribution activates and increases with temperature. Beside this fact its behaviour is known to be very close to the one of an ideal gas. Different simulations have been carried out to find out the differences occurring in the optimization process as the gas is treated as an ideal and a real one.

```
%% THERMODYNAMIC PARAMETERS
% Number of Stages
nst=3;
% 1 ->ideal gas, 2 ->real gas
igasp=1;
% Fluid (check refprop folder)
fluid='nitrogen';
% Mass flow rate [kg/s]
amass=10;
% Total inlet temperature [K]
tin=1023;
% Total inlet pressure [Bar]
pin=5;
% Pressure ratio inlet/outlet (beta)
raptot=5;
% Revolution per minute (rpm)
rpm=8000;
% Constant specific heat ratio (if igasp=1)
gampt=[1.3391 1.3391 1.3391];
```

In the ideal cases three different settings are taken in to account, a constant specific heat ratio throughout the whole turbine equal to the value it would have at the inlet temperature and very low pressure, an intermediate value and a third value calculated at outlet conditions. A fourth try was run adopting different specific heat ratios for each stage, equal to the one adopted in the previous three optimizations.

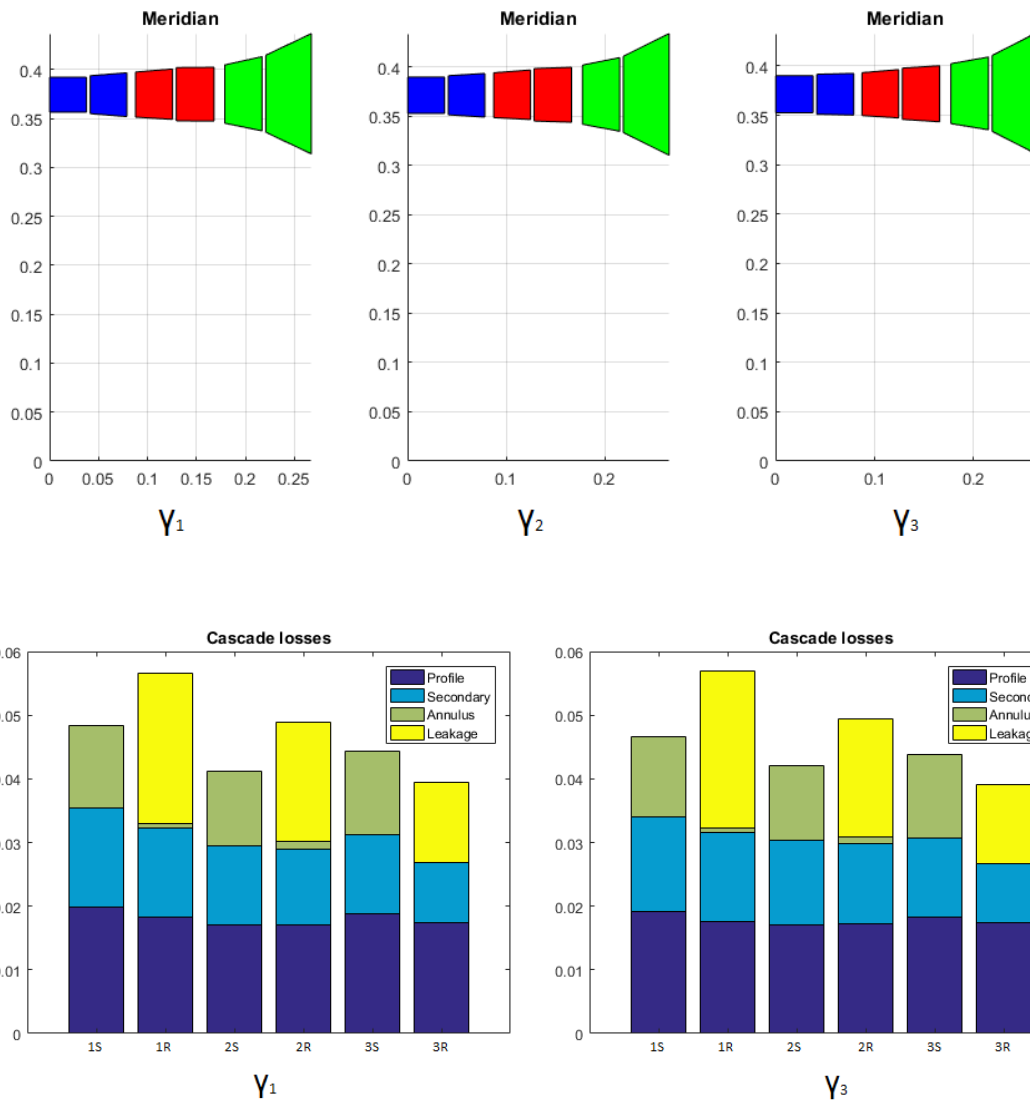
For the real gas optimization both MATLAB and the FORTRAN version of Axtur are analysed to show the differences.

The three values of gamma specified in the first three cases are: $\gamma_1 = 1.3391$, $\gamma_2 = 1.3548$, $\gamma_3 = 1.3738$. The slightly larger value taken at the isentropic outlet temperature causes a larger temperature drop in the turbine where that value is kept constant throughout the stages and a consequent smaller volume flow rate at the last stage outlet. The difference is limited to 2% and doesn't affect much the final solution.

As it can be seen from the meridian planes, the designs are practically the same for all the solutions. In the first case, where the volume flow rate is higher a very slightly larger kinetic energy loss can be found at the outlet of the last stage. That said all the turbines analysed with nitrogen shows the same efficiency of 91.4% with the results from the real

gas in MATLAB and FORTRAN optimizing the turbine to the exact same efficiency down to the third decimal digit (although this kind of precision is just a coincidence it shows how, for a gas like nitrogen, the real gas effect calculated by the FORTRAN version of Axtur are very close to the one in MATLAB).

Reported are the meridian plane and the cascade losses for the first three cases.

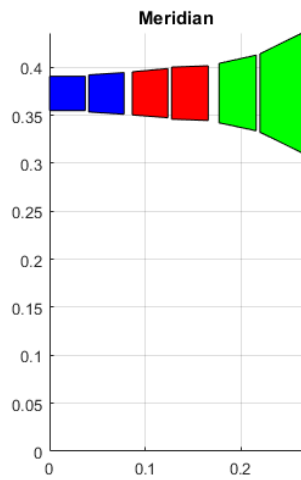


In the first stage, where the volume flow ratio is slightly larger, a higher Mach number is achieved at the end of the first stator to allow the passage of the fluid.

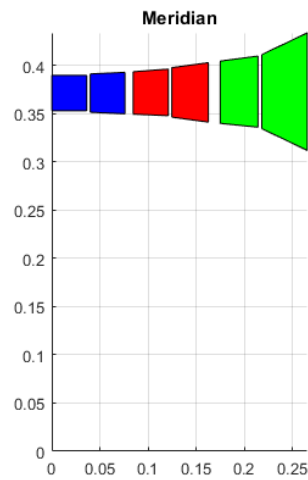
The case where three different values of gamma are adopted for every stage is the one that better simulate the volumetric behaviour found in the real gas. The results are again

very similar and show no significant differences in geometry and losses. Considering the different method that are adopted to compute the evolution of the gas in the ideal case and the real one and the number of variable to be optimized, the slightly differences could be attributed to the optimization rather than to real differences in the fluid properties.

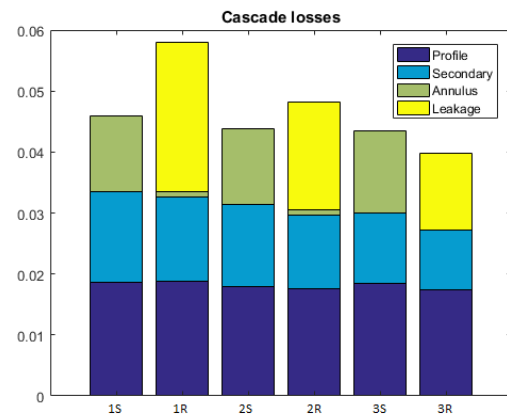
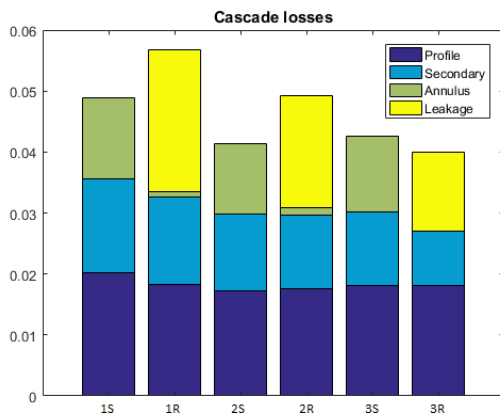
Reported are the meridian planes and the losses for the case with three different gammas and the real one.



3 gammas



Real



Reported in the table below are the value of beta, kis, reaction grade, h/d ratio for the rotor and the outlet isentropic mach. Where a value is larger in a stage is than lower in the next, bringing the optimization to the same result.

Furthermore, it must be noted that the strong assumption of total independence between all the losses is made for the calculation, making the results meaningful for discussion when an appreciable difference occur.

Stage		γ_1	γ_2	γ_3	3 gammas	Real FORTRAN	Real MATLAB
1	beta	1.6621	1.6518	1.6514	1.6997	1.6759	1.6738
	kis	2.94241	2.95394	2.9429	3.08956	3.02601	3.02952
	reaction	0.41312	0.4204	0.43426	0.40862	0.41762	0.44153
	h/d rotor	0.05978	0.05971	0.05679	0.05827	0.06372	0.05786
	mach out	0.60759	0.60439	0.60742	0.62183	0.5921	0.62593
2	beta	1.7778	1.7937	1.7838	1.7964	1.7434	1.7235
	kis	2.94958	3.03125	2.98134	3.0077	2.86492	2.83316
	reaction	0.48139	0.47557	0.46929	0.48116	0.44908	0.44794
	h/d rotor	0.07331	0.07496	0.07632	0.07632	0.08291	0.08286
	mach out	0.69077	0.69168	0.68099	0.69555	0.6365	0.64919
3	beta	1.7666	1.7637	1.7729	1.715	1.7713	1.8004
	kis	2.57019	2.58287	2.57391	2.43417	2.53255	2.67328
	reaction	0.4027	0.39561	0.39606	0.42558	0.40659	0.38913
	h/d rotor	0.16364	0.16586	0.16258	0.16786	0.15749	0.16353
	mach out	0.63912	0.63225	0.63198	0.62659	0.6164	0.6382

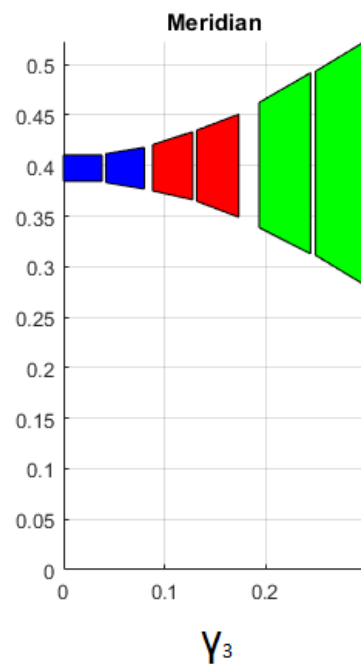
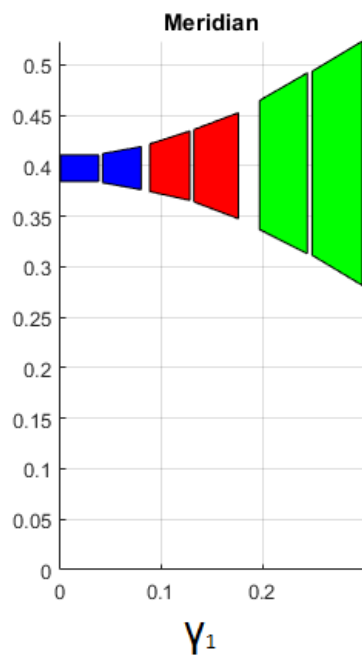
In the next paragraph an ORC fluid with strong real gas effect is analysed.

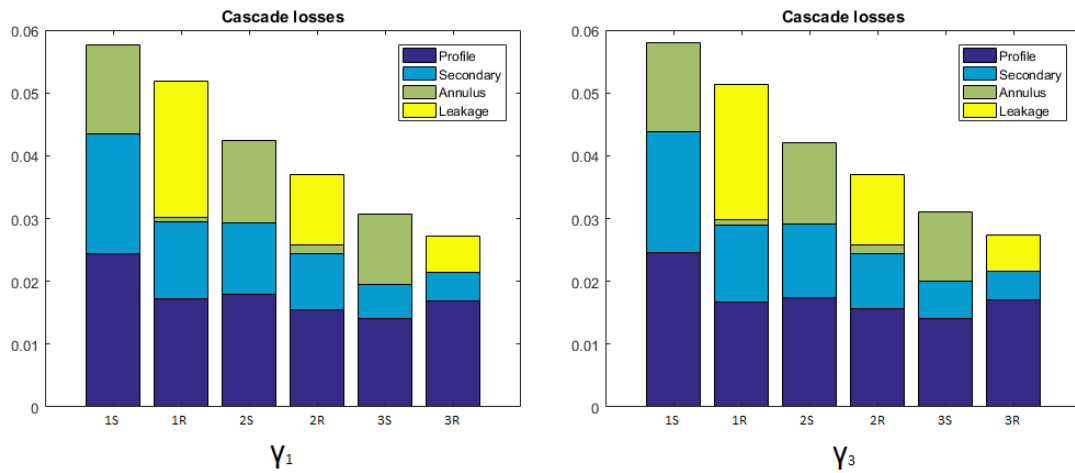
3.1.3 Octane

Octane is heavy hydrocarbon with a complex molecule. Its specific heat ratio approaches 1 while the temperature rise and all the vibrational contributions are activated. A cycle starting from a condition very close to the saturation line is taken in to account. In this condition the fluid shows strong real gas effects, for instance, the compressibility factor, that is assumed equal to 1 for an ideal gas, is very far from this value in the real

```
%% THERMODYNAMIC PARAMETERS
% Number of Stages
nst=3;
% 1 ->ideal gas, 2 ->real gas
igasp=2;
% Fluid (check refprop folder)
fluid='octane';
% Mass flow rate [kg/s]
amass=63;
% Total inlet temperature [K]
tin=536.15;
% Total inlet pressure [Bar]
pin=15.4;
% Pressure ratio inlet/outlet (beta)
raptot=30;
% Revolution per minute (rpm)
rpm=4900;
% Constant specific heat ratio (if igasp=1)
gampt=[1.03 1.03 1.03];
```

case: $Z_{in}=0.611$, this lead to a big difference in volume flow rate of the first stage. As in the previous case different approaches are analysed. Due to the very low variation in temperature and consequently in gamma, the three cases in which a constant value of the specific heat ratio is adopted throughout the whole turbine shows no differences at all.

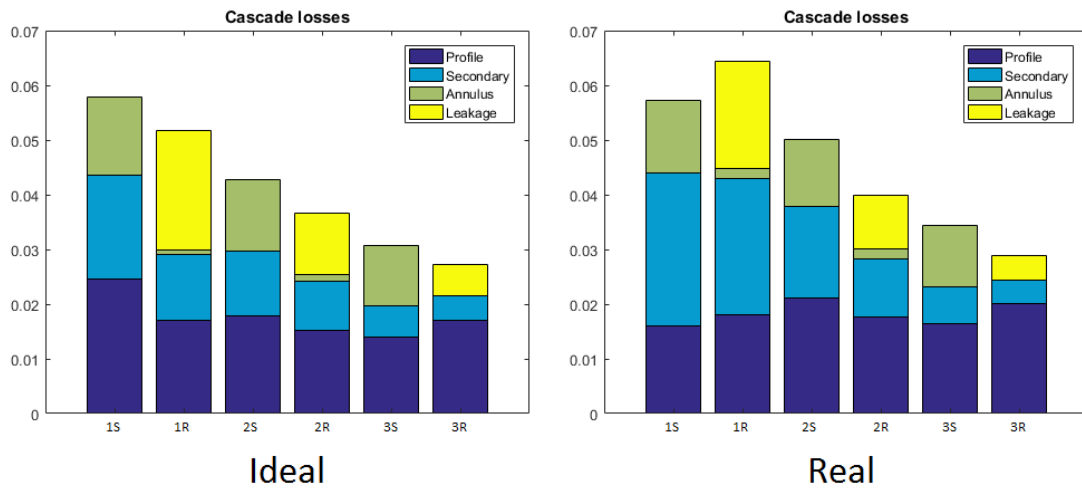
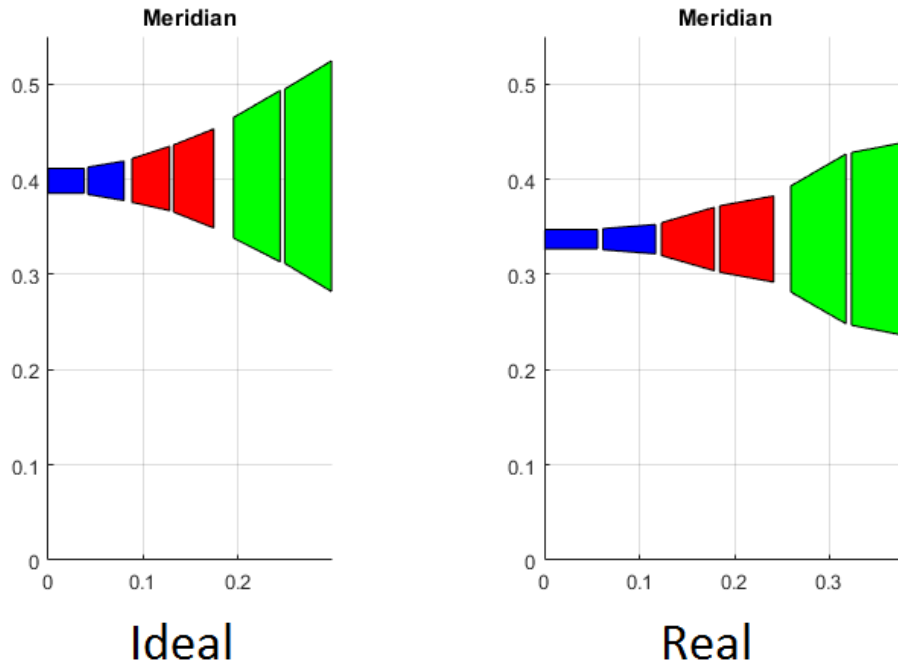




Listed in the table are the main parameters in the two optimization process. No appreciable difference can be found as expected from the very low variation in the value of gamma, ranging from 1.0283 to 1.0314

Stage		γ_1	γ_3
1	beta	3.4188	3.4128
	kis	2.26255	2.25993
	r^*	0.38408	0.38018
	h/d stator	0.03282	0.03262
	h/d rotor	0.05354	0.05187
	mach stator	1.21496	1.21701
	mach rotor	1.00422	0.99933
2	beta	2.8219	2.8574
	kis	1.8347	1.85403
	r^*	0.45378	0.45816
	h/d stator	0.08572	0.08415
	h/d rotor	0.13104	0.12728
	mach stator	1.04965	1.05067
	mach rotor	1.00034	1.01107
3	beta	3.3795	3.3588
	kis	2.06714	2.04914
	r^*	0.51797	0.52087
	h/d stator	0.22248	0.22254
	h/d rotor	0.30052	0.30044
	mach stator	1.0672	1.06019
	mach rotor	1.15091	1.15109

The real case, on the other hand shows remarkable differences compared to the ideal one.

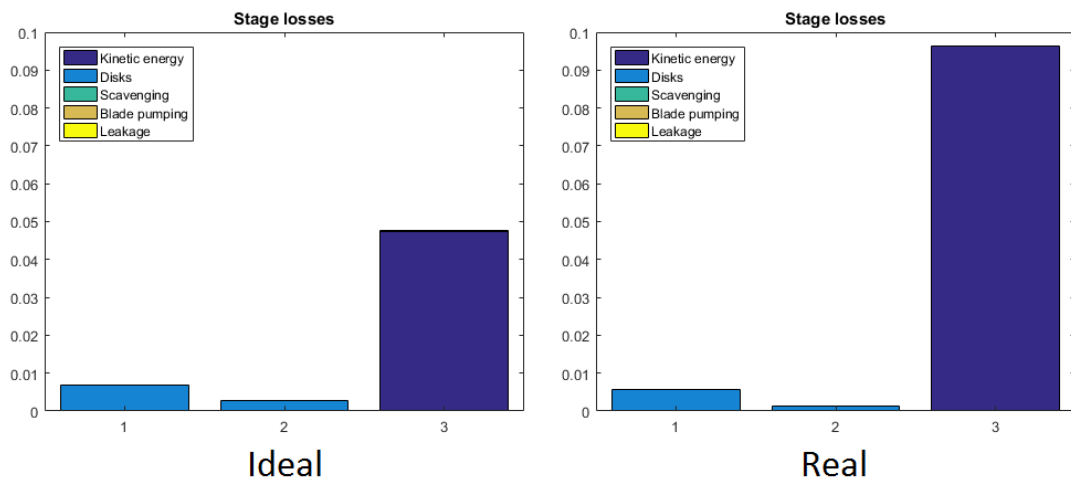


In the real case, due to the very low compressibility factor in the inlet conditions, a very low blade height would be achieved if the mean diameter had to be kept constant. This would lead to very high secondary losses and a very low efficiency. To try to cope with this problem the optimization process reduces the mean diameter of the machine to increase the blade height and minimize secondary losses. As seen in the following table the difference in flow rates of the first stage is remarkable, while as the fluid expands and

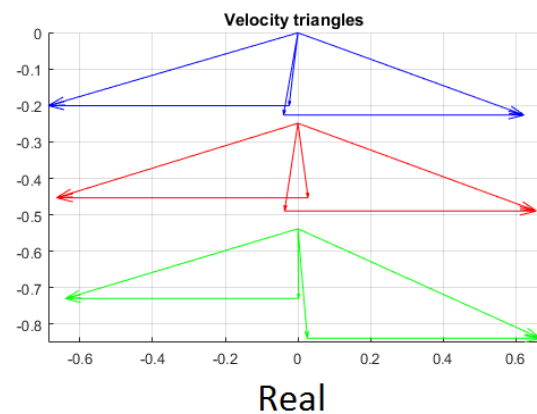
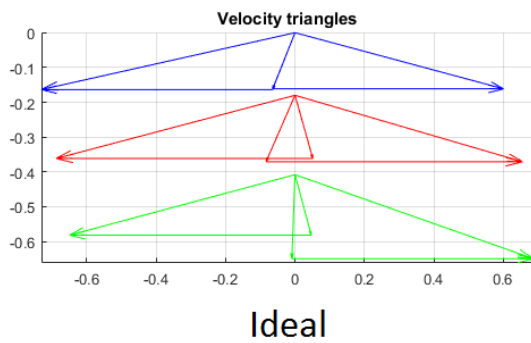
approaches ideal gas condition, the two values start to merge. This can be appreciated from the cascade losses diagram where the differences reduce as the fluid approaches outlet conditions.

REAL				
Vin stator	0.9864	4.3768	12.6917	m ³ /s
Vin rotor	2.6756	7.9409	22.5673	m ³ /s
Vout rotor	4.7573	13.6677	41.8277	m ³ /s
IDEAL				
Vin stator	1.5994	5.0623	13.4812	m ³ /s
Vin rotor	3.3145	8.8751	23.582	m ³ /s
Vout rotor	5.2594	14.1043	43.8307	m ³ /s

Another problem that the optimization process has to manage while searching for the best efficiency is the loss related to the kinetic energy at the exhaust. As it tries to reduce the secondary losses to the first stage by reducing the mean diameter and increasing blade height, the kinetic energy to the exhaust greatly increase having to discharge almost the same volume flow rate of the ideal case from a much smaller area.



The axial component in the velocity triangles significantly increases to deal with the increased volume flow rate.



Overall the efficiency achieved by the turbine having to deal with the real fluid it's about 2.5% lower than the one obtained by the ideal process.

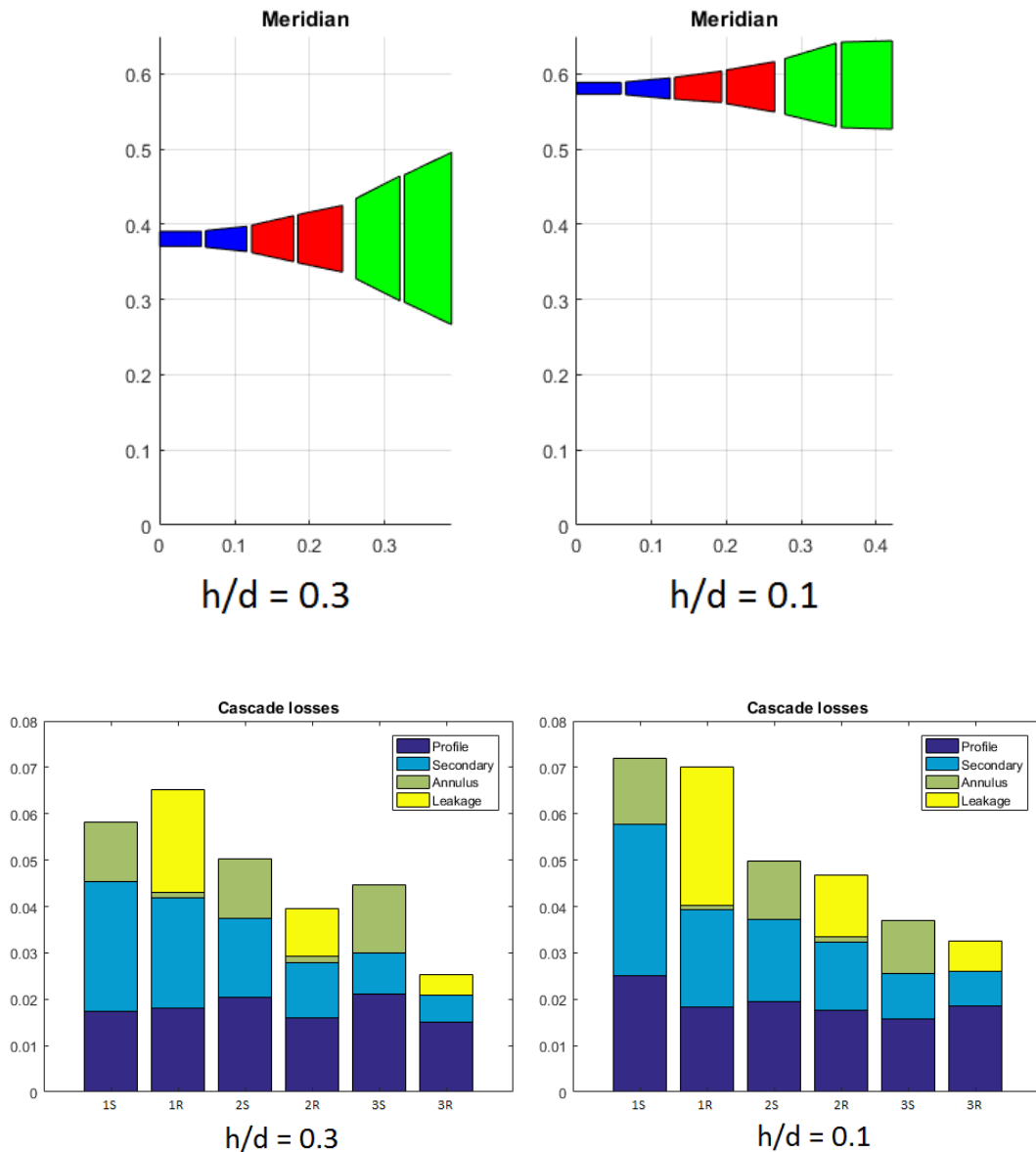
In the first two stages of the real turbine a higher reaction grade is chose to reduce deflection and increase the velocity needed to discharge the greater volume ratio. For the same reason a higher kis is chosen on the first stage to reduce the mean diameter to increase the h/d ratio. Even if a higher h/d ratio could reduce the secondary losses to the first stage, the limit value of 0.3 is reached to the last stage, preventing the optimization to further increase the kis of the first stage.

Stage		Ideal	Real MATLAB
1	beta	3.3937	3.5074
	kis	2.24702	2.50816
	r*	0.38326	0.44383
	h/d stator	0.03272	0.02993
	h/d rotor	0.05213	0.04584
	mach stator	1.21214	1.24978
	mach rotor	0.99995	1.09723
2	beta	2.8659	2.9968
	kis	1.86024	2.48151
	r*	0.44823	0.49082
	h/d stator	0.08412	0.09903
	h/d rotor	0.12978	0.13418
	mach stator	1.0629	1.07119
	mach rotor	1.00236	1.07661
3	beta	3.3611	3.3201
	kis	2.05553	2.74406
	r*	0.52142	0.52025
	h/d stator	0.22315	0.26435
	h/d rotor	0.30049	0.30146
	mach stator	1.06093	1.06755
	mach rotor	1.15207	1.15666
	Overall efficiency	90.05%	87.62%

This prove how the effects related to

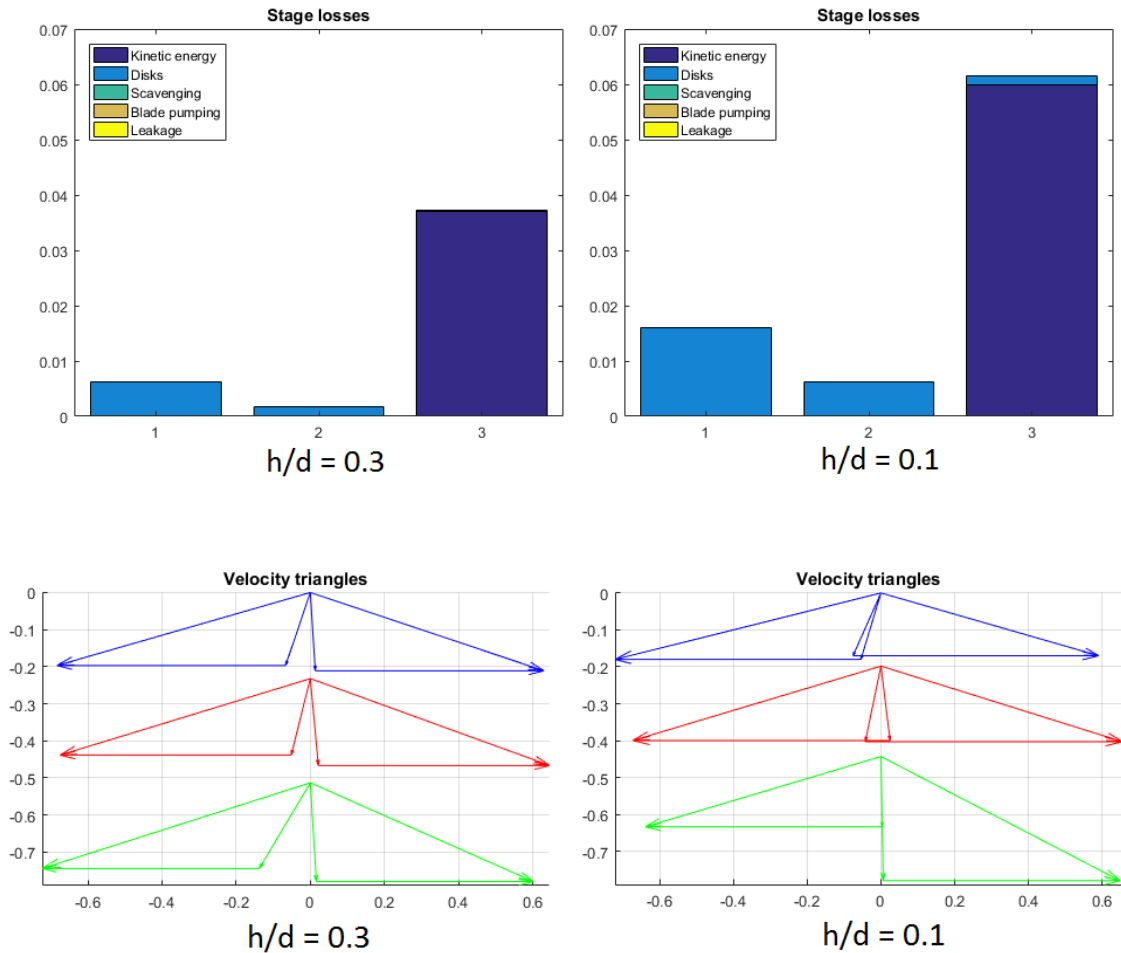
real gas behaviour can vastly modify the optimization process of gas turbine heaving to deal with almost twice the volumetric expansion ratio. ($V_{ratio_ideal} = 27.4$, $V_{ratio_real} = 42.4$)

Since the program simulate a 1D problem with all the properties defined at a middle blade height, an h/d ratio of 0.3 (suitable for twisted blades) underestimate the effects due to different peripheral speed from hub to tip. A further try reducing the h/d ratio to 0.1 (limit for prismatic blades) was carried out to analyse the differences in the optimization process.



As shown from the meridian plane, the mean diameter increase trying to reduce secondary losses by increasing blade height. Optimal rotating speed also reduces to 3000 rpm to reduce the peripheral velocity. The overall efficiency is lower compared to the h/d 0.3 because secondary losses are higher on all stages due to lower blade height. The

kinetic energy to the exhaust increases having to discharge the same volume flow rate from a smaller area and outlet velocity shows no tangential component trying to maximise the energy recovered from the diffuser.



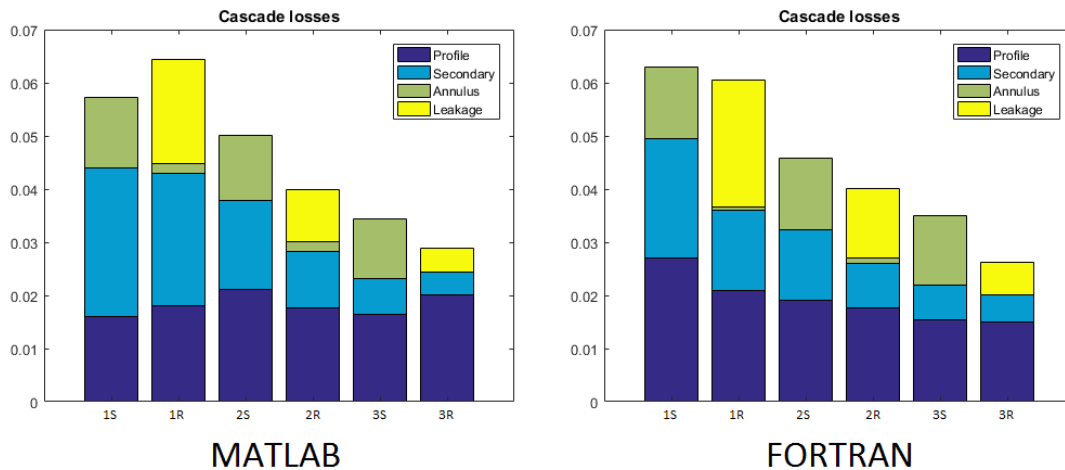
As the velocity triangles show, the final solution greatly relies on the kinetic energy recovered by the diffuser. A run without diffuser converge to a lower overall efficiency with higher profile and secondary losses on all stages, having to increase the load on the blades to try to mitigate the higher kinetic energy lost to the exhaust.

Kis on the first stage is lower to achieve a higher mean diameter and Mach numbers are higher to allow the flow to be discharged from a smaller outlet section.

Stage		h/d = 0.3	h/d = 0.1
1	beta	3.4582	3.5337
	kis	2.62852	2.2553
	r*	0.43723	0.37803
	h/d stator	0.02635	0.01345
	h/d rotor	0.04408	0.02394
	mach stator	1.25033	1.3063
	mach rotor	1.07644	1.00582
2	beta	2.9094	2.7767
	kis	2.56845	2.06009
	r*	0.4531	0.45917
	h/d stator	0.08022	0.03582
	h/d rotor	0.11651	0.05747
	mach stator	1.09383	1.06395
	mach rotor	1.02871	1.0087
3	beta	3.3198	3.32
	kis	2.9156	2.42663
	r*	0.37978	0.52029
	h/d stator	0.21784	0.09474
	h/d rotor	0.30041	0.10026
	mach stator	1.21292	1.06754
	mach rotor	1.03555	1.14917
	Overall efficiency	89.49%	87.36%
	rpm	4200	3000

As a final check the same process is optimized in FORTRAN running Axtur with real gas hypothesis.

As is shown from the plot of the losses through the cascades there is a considerable difference in the first stage that decreases in the later stages. In FORTRAN the values that are set at the beginning of the stage are kept constant both for stator and rotor and the compressibility factor is linearly interpolated at an average pressure. Since the expansion starts very close to the saturation line and the properties varies considerably as the fluids moves from that point, it is hard to define a middle condition that satisfy both stator and rotor of the first stage.



Having to deal with a lower volume flow rate in the first stage the results shows lower secondary losses and higher leakage due to a lower Mach number in the first stage.

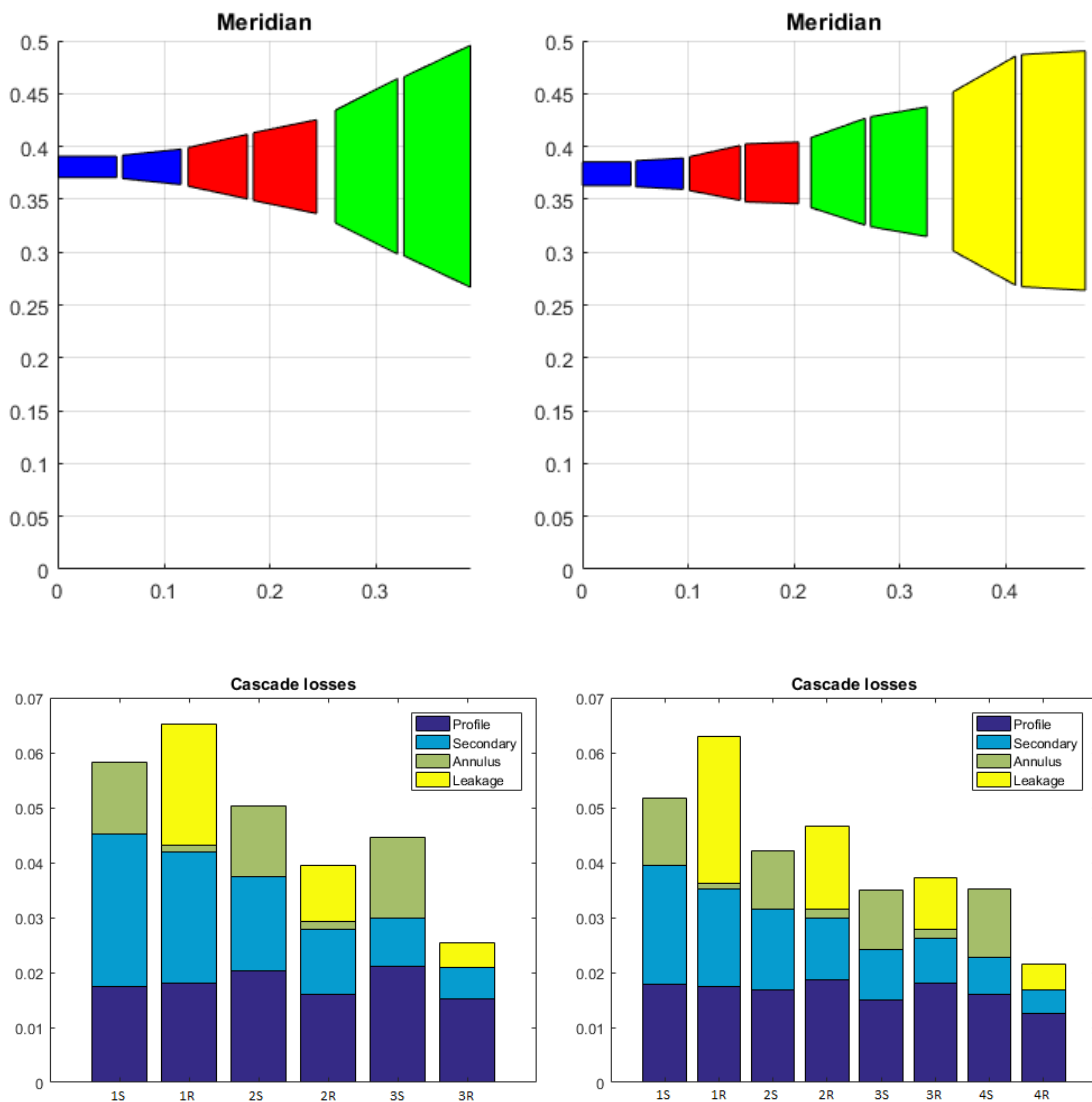
A fine-tuning of the properties chosen for the real optimization can in theory reduce the differences and make it more accurate. This shows how the integration of REFPROP in Axtur greatly simplifies the usage of this tool, requiring no input from the user and making a reliable and accurate prediction of the evolution of the fluid properties within the machine.

3.2 Effects increasing number of stages

One of the limitations of the original FORTRAN version of Axtur was the limit of 30 variables that the optimization was able to handle. Because each stage required nine variables to be optimized, the maximum number of stages was limited to three. During the conversion to MATLAB this limit was removed letting the program run optimization for any number of variables. Running time usually stays within a minute for optimizations up to five/six stages (It is highly variable depending on the starting parameters).

An analysis is carried out optimizing the previous octane turbine with a number of stages ranging from one to five. Reported are the meridian planes and the cascade losses through the stages.

Moving from three to four stages shows an increment in the overall efficiency from 89.4% to 90.5%. Due to the high volumetric expansion ratio happening in the first stage of the three-stage turbine, adding a fourth stage allow a reduction of load on that stage. The optimal speed reduces and the blade height in the first stage increase reducing the secondary losses and the Mach numbers. Leakage of the first rotor increase due to the slower flow but is compensated by the overall reduction of secondary losses in all the other stages.



Stage		3-stage turbine	4-stage turbine
1	beta	3.4582	2.6614
	kis	2.62852	2.3067
	r*	0.43723	0.47404
	h/d stator	0.02635	0.03002
	h/d rotor	0.04408	0.03947
	mach stator	1.25033	1.10893
	mach rotor	1.07644	0.99354
	Vin stator	0.9754	0.9754
	Vout rotor	4.6692	3.5015

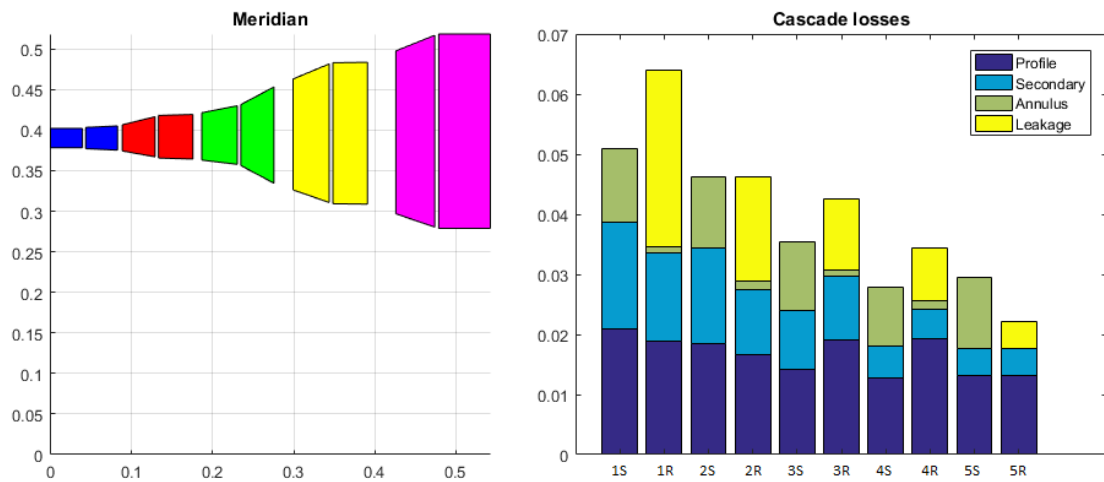
Because the mean diameter and the peripheral speed are both function of the enthalpy drop of the stage and the kis:

$$u = \sqrt{\frac{2\Delta H_{stage}}{k_{is}}}$$

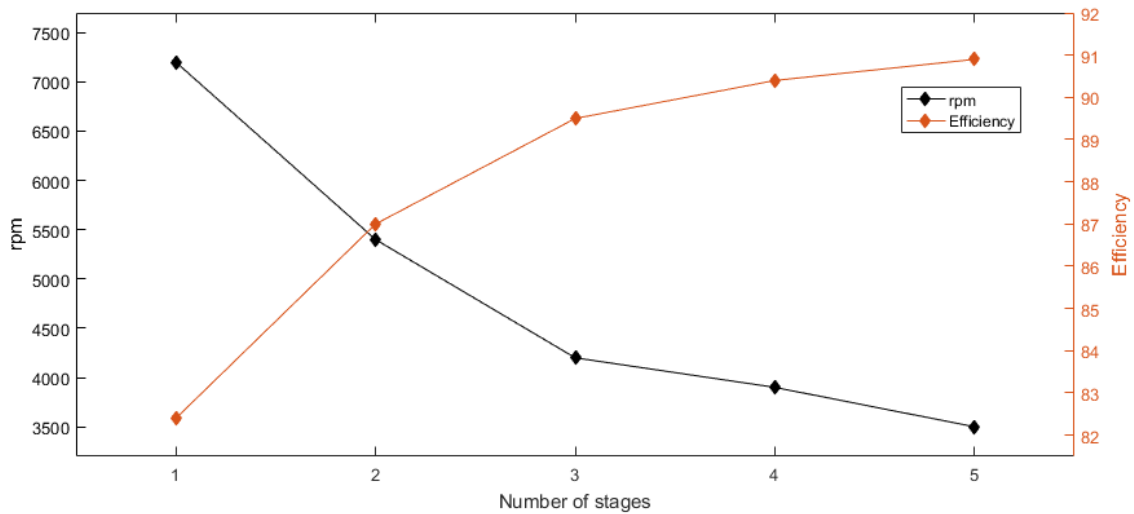
$$d = \frac{u}{\omega}$$

a reduction on the stage enthalpy drop, due to the same overall enthalpy split on more stages, allow a reduction of the kis of the first stage to maintain an optimal peripheral velocity. This results in lower kis throughout all the stages, higher blades and lower secondary losses.

Adding a fifth stages do not improve much the overall efficiency raising it to 90.9%. Moreover, the optimization reaches the non-linear constraint on the minimum ratio between inlet and outlet blade (equal to one to avoid convergence) meaning that it wanted to move in that direction to increase the efficiency. On the four-stage turbine all the stages shows a degree of reaction close to 50% and a nearly transonic flow, making it a well-balanced turbine. A fifth stage adds no benefit to the efficiency and makes the turbine more expensive. The only advantage in choosing this configuration is that the optimal speed is reduced to 3600 rpm allowing it to be coupled directly to a generator without the need of a gearbox (in the American market where the frequency is equal to 60Hz).



Plotted are the trends of efficiency and optimal rotating speed for turbines ranging from one to five stages. As expected the two functions are continuous and as the efficiency grows the optimal rotating speed reduces to maintain an optimal peripheral speed lowering the k_{is} on all stages.



4. Conclusions

The MATLAB version of Axtur proves to be a powerful and reliable tool to estimate the geometry and the thermodynamic processes happening in a gas turbine. The improvement added by the integration of REFPROP is of great importance when dealing with fluids that shows strong real gas behaviours. The conversion itself, switching from a lower level language like FORTRAN to a high level one, that is MATLAB, allows further development on the program to be carried out in a much easier way. This also introduces a downside, the time needed for the optimization. In FORTRAN, running Axtur on a 2011 machine takes less than a second for any optimization to converge and provide a solution. On MATLAB this process takes much longer, especially when run in real gas mode. The preallocation process, that needs some 110,000 calls to REFPROP, takes between 30 and 60 seconds depending on the fluid. Overall, the time needed for a multi stage (up to five/six stages in real gas mode) simulation usually stays within a couple of minutes, making it much slower than FORTRAN but still managing to provide a solution in a reasonable amount of time.

4.1 Further developments

There are many aspect of the program that can be improved to allow Axtur to be a much more reliable and complete tool when dealing with axial turbine calculations.

- A smart initialization process could be implemented to allow the program to avoid crashing when the first iteration returns NaN as a solution.
- New correlation could be added to simulate twisted blade and non-optimized velocity triangles on the hub and tip of the blades.
- Off-design performance prediction.

- From the optimization of the code point of view a revision on how the preallocated variables are fed to the “funzot” and “funzot_real” scripts must be carried out. Currently half of the time needed for the optimization process (preallocation excluded) is wasted to access the MAT file in which the preallocated variables are stored. This can be fixed feeding the preallocation output as a struct instead of saving it in a file.

References

- [1] Macchi E., "Closed cycle turbine", 1977
- [2] Astolfi M. "Principi di dimensionamento di Turbomacchine", 2015
- [3] Craig H.R.M., Cox H.J.A., "Performance estimation of axial flow turbines", 1970
- [4] Wei N., "Significance of Loss Models in Aerothermodynamic Simulation for Axial Turbines", 2000
- [5] Macchi E., Astolfi M., "Organic Rankine Cycle (ORC) Power Systems", 2016

List of figures

- Coordinate system for axial turbine, pag 2
- Optimization mesh, pag 3
- CFD simulation, pag 4
- Turbine geometry, pag 8
- Velocity triangles, pag 10
- Optimization flowchart, pag 21
- Axtur flowchart, pag 25
- Graphical output example, pag 28
- Conversion check, pag 29
- Optimization function comparison, pag 30
- Optimization results, pag 31
- Kis=2, kis=5 optimization, pag 33
- Neon real vs ideal, pag 43
- Stage losses vs rotating speed, pag 44
- Single stage optimization, pag 45
- Nitrogen ideal optimization, pag 47
- Nitrogen real optimization, pag 48
- Octane ideal optimization, pag 50
- Octane real optimization, pag 52
- Octane FORTRAN vs MATLAB losses, pag 55
- 3 vs 4 stages optimization, pag 57
- 5 stages optimization, pag 58

